

В. С. СИЗИКОВ

# ОБРАТНЫЕ ПРИКЛАДНЫЕ ЗАДАЧИ и MatLab

*РЕКОМЕНДОВАНО*

*Учебно-методическим объединением вузов РФ  
по образованию в области приборостроения  
и оптоэлектроники для студентов  
высших учебных заведений, обучающихся  
по направлению подготовки 200100 —  
"Приборостроение" и специальности 200101 —  
"Приборостроение"*

САНКТ-ПЕТЕРБУРГ · МОСКВА · КРАСНОДАР  
2011

**Сизиков В. С.**

**С 34** Обратные прикладные задачи и MatLab: Учебное пособие. — СПб.: Издательство "Лань", 2011. — 256 с.: ил. (+ вклейка, 8 с., + CD) — (Учебники для вузов. Специальная литература). Исправл. эл. вариант.

**ISBN 978-5-8114-1238-9**

Книга посвящена применению аппарата интегральных уравнений (ИУ) и программных средств системы MatLab к решению ряда прикладных задач томографии, иконики и спектроскопии. Изложены понятия прямых и обратных задач, задачи рентгеновской компьютерной томографии и ЯМР-томографии, задачи иконики — реконструкции искаженных (смазанных, дефокусированных и зашумленных) изображений и спектроскопии. Обратные задачи описаны, как правило, интегральными уравнениями Фредгольма I рода, задача решения которых некорректна, поэтому уравнения решаются методом регуляризации Тихонова или методом параметрической фильтрации Винера. Методы и алгоритмы доведены до программ в версии MatLab 7. Приведены листинги программ и результаты обработки модельных и реальных данных. Применительно к задаче иконики изложены как известные методы восстановления изображений, так и разработанная автором методика под названием "усечение–размытие–поворот". Изложена также новая методика решения обратной задачи спектроскопии для случая непрерывного спектра — способ моделирования и для случая дискретного спектра — алгоритм интегральной аппроксимации.

*Для студентов, магистрантов, аспирантов и преподавателей вузов, а также для специалистов по прикладной и вычислительной математике.*

Работа выполнена при поддержке РФФИ (грант № 09-08-00034а).

ББК 22.161.6я73

**Рецензенты:**

*В. М. ИВАНОВ* — доктор физико-математических наук, профессор, зав. кафедрой информационных технологий в дизайне ИМОП СПбГПУ;

*И. П. ГУРОВ* — доктор технических наук, профессор, зав. кафедрой компьютерной фотоники СПбГУ ИТМО.

**Обложка***А. В. ПАНКЕВИЧ*

*Охраняется законом РФ об авторском праве.*

*Воспроизведение всей книги или любой ее части  
запрещается без письменного разрешения издателя.*

*Любые попытки нарушения закона  
будут преследоваться в судебном порядке.*

© Издательство "Лань", 2011

© В. С. Сизиков, 2011

© Издательство "Лань",

художественное оформление, 2011

# ОГЛАВЛЕНИЕ

<b>Предисловие</b> .....	6
<b>Список сокращений</b> .....	8
<b>Введение. Прямые и обратные задачи</b> .....	11
Примеры прямых задач (11). Примеры обратных задач (15). Две схемы измерений и обработки (16). Контрольные задания и вопросы (18).	
<b>Глава 1. ЗАДАЧИ КОМПЬЮТЕРНОЙ ТОМОГРАФИИ</b> .....	19
<b>1.1. Рентгеновская компьютерная томография (РКТ)</b> .....	21
Недостатки обычной рентгенографии (21). Техническая идея РТ (21). Пять поколений рентгеновских томографов (22). РКТ-сканер (25). Закон Бера (26). Преобразование и уравнение Радона (26). Методы решения уравнения Радона (27). Теорема Брейсуэлла о сечении спектра (27). Метод преобразования Фурье (28). Метод свертки и обратной проекции (три варианта) (29). Приведение уравнения Радона к уравнению Фредгольма I рода (32). Решение уравнения Фредгольма методом ПФ с регуляризацией Тихонова (33). Историческая справка (35). Численные иллюстрации (35). Задачи с неполными данными (37). Области применения РКТ (37). Контрольные задания и вопросы (37).	
<b>1.2. Ядерно-магнитно-резонансная томография (МРТ)</b> .....	39
Историческая справка (39). Поведение магнитных моментов ядер без поля и в поле (41). Уравнение Лармора и ларморова прецессия (41). Ансамбль протонов (43). Движение магнитного момента протона в постоянном и переменном магнитных полях (44). Уравнения Блоха (45). Некоторые численные данные (49). ЯМР-сигнал (51). Основные импульсные последовательности (52). Градиентные поля (56). Реконструкция ЯМР-изображений (основные принципы) (58). Сечение и слой (60). Методика Хоулта (62). Конкретные методы и схемы реконструкции ЯМР-изображений (62). Влияние неоднородности полей на разрешающую способность ЯМР-изображений (65). Области применения ЯМР-томографии (66). Примеры реконструкции ЯМР-изображений (66). Контрольные задания и вопросы (66).	
<b>Глава 2. РЕКОНСТРУКЦИЯ ИСКАЖЕННЫХ ИЗОБРАЖЕНИЙ</b> ..	68
<b>2.1. Объекты изображений, системы наблюдений, типы искажений</b> ..	68
Объекты изображений (68). Системы наблюдений (68). Типы искажений изображений (69). Типы изображений и классы чисел в системе MatLab (69). Соглашение о координатах (70). Примеры искаженных изображений (71). Контрольные задания и вопросы (72).	
<b>2.2. Аберрации оптических систем</b> .....	72
Типы аберраций (72). Сферическая аберрация (72). О космическом телескопе "Хаббл" (73). Контрольные задания и вопросы (74).	

<b>2.3. Реконструкция смазанных изображений (теория)</b> .....	74
Прямая задача (74). Обратная задача устранения смаза (76). Вывод интегрального уравнения (76). Метод дифференцирования (78). Преобразование уравнения (2.8) к уравнению типа свертки (78). Об использовании "граничных условий" (BCs) (80). Прием усечения смазанного изображения (82). Схема для понижения эффекта Гиббса (84). Обратная задача (86). Метод преобразования Фурье (86). Метод псевдоинверсной фильтрации (87). Метод ПФ с регуляризацией Тихонова (88). Метод конечных сумм с регуляризацией Тихонова (89). Метод фильтрации Винера (93). Контрольные задания и вопросы (94).	
<b>2.4. Реконструкция смазанных и зашумленных изображений (численные иллюстрации)</b> .....	94
Программы в системе MatLab (94). Текстовое изображение (101). Прямая задача без зашумления (102). Обратная задача без зашумления и с точной ФРТ (103). Вариант 1 (103). Вариант 2 (106). Варианты 3, 4 (108). Использование функции <code>edgetaper.m</code> (113). Вариант 5 (114). Зашумленность смазанного изображения и погрешность ФРТ (117). Портретное изображение (125). Изображение со смазыванием и гауссовым шумом (126). Изображение со смазыванием и импульсным шумом (130). Сравнение с результатами других публикаций (137). Замечания (137). Быстрый алгоритм реконструкции смазанных изображений (137). Изображения, смазанные под углом, прямая и обратная задачи (139). Обработка цветных изображений (146). Обработка реальных смазанных изображений (151). Контрольные задания и вопросы (154).	
<b>2.5. Реконструкция дефокусированных изображений</b> .....	156
Постановка задачи (157). Вывод основного соотношения в простейшем случае (158). Приведение основного уравнения к стандартной форме (160). Численные алгоритмы решения прямой задачи (162). Прием усечения дефокусированного изображения (163). Прием размытия краев дефокусированного изображения (167). Собственные $m$ -функции для моделирования дефокусирования изображений (169). Решение обратной задачи реконструкции дефокусированных изображений (172). Собственные $m$ -функции для реконструкции дефокусированных изображений (177). Примеры реконструкции дефокусированных изображений (178). Метод итераций с квадратурами (186). О сравнительных возможностях редактора PhotoShop (195). Контрольные задания и вопросы (197).	
<b>Глава 3. ОБРАТНЫЕ ЗАДАЧИ СПЕКТРОСКОПИИ</b> .....	199
<b>3.1. Введение в спектральный анализ</b> .....	199
Спектральный анализ и приборы (199). Типы спектров (199). Типы спектрального анализа (201). Области применения спектрального	

---

анализа (201). Контрольные задания и вопросы (203).	
<b>3.2. Математическое описание задачи спектроскопии в случае непрерывного спектра</b> .....	203
Основные понятия и определения (203). Задача редукции к идеальному спектральному прибору (205). Математическая формулировка задачи спектроскопии (205). Метод квадратур с регуляризацией Тихонова (207). Численная иллюстрация с использованием способа модельных примеров (211). Контрольные задания и вопросы (216).	
<b>3.3. Задача спектроскопии в случае дискретного спектра</b> .....	217
Математическое описание задачи (217). О решении СЛНУ (219). Алгоритм интегральной аппроксимации решения СЛНУ (220). Численный пример (221). О некоторых особенностях программирования в системе MatLab (225). Критерий Рэлея и редукционная проблема Рэлея (227). Контрольные задания и вопросы (230).	
<b>Литература</b> .....	232
<b>Предметный указатель</b> .....	245

## ПРЕДИСЛОВИЕ

Данная книга является учебником для студентов, магистрантов, аспирантов и преподавателей вузов, а также монографией для научных сотрудников в области прикладной и вычислительной математики, физики и информатики (программирования).

В книге изложены некоторые, достаточно востребованные прикладные задачи, а именно, задачи рентгеновской компьютерной томографии (РКТ), ядерно-магнитно-резонансной томографии (МРТ), иконики (науки об обработке искаженных изображений) и спектроскопии. Что объединяет эти, казалось бы, весьма различные задачи? Их объединяет весьма похожий математический аппарат. Все эти задачи описаны, как правило, интегральными уравнениями (ИУ) I рода Фредгольма, а также Вольтерра и Радона. А задача решения таких уравнений является некорректной (существенно неустойчивой). Поэтому в книге для их устойчивого решения используются методы регуляризации Тихонова и параметрической фильтрации Винера.

Автор на собственном опыте убедился, что методы должны быть доведены до численных алгоритмов и программ на каком-то языке программирования — только в этом случае можно говорить о достаточно полном изложении прикладных задач. Опыт программирования на различных языках (Fortran, Pascal, C/C++ и др.), а также в системах программирования (MathCAD, MatLab и др.) показал, что одним из самых удобных и эффективных является программирование в системе MatLab, особенно, программирование задач обработки изображений. Поэтому все задачи в книге запрограммированы в системе MatLab, причем приведены полностью (на CD-ROM) листинги всех головных программ и m-функций.

В главе 1 при изложении задачи РКТ акцент сделан на вопросе о решении ИУ Радона методом свертки и обратной проекции, а также на приведении уравнения Радона к 2-мерному ИУ Фредгольма I рода типа свертки. В изложении МРТ основное внимание уделено реконструкции ЯМР-изображений (метод Кумара–Велти–Эрнста и т.д.).

Наибольшей по объему является глава 2, посвященная одному из разделов иконики — реконструкции искаженных (смазанных, дефокусированных и зашумленных) изображений. Как известно, в многочисленных работах Gonsales, Woods, Eddins, Nagy, Donatelli и др. широко используется прием (алгоритм, способ, методика), неудачно названный "Boundary Conditions" (BCs) — "граничные условия". Этот прием используется для экстраполяции значений интенсивности на изображении за границы изображения. Аналогичные способы доопределения изложены в книге Л.П. Ярославского [143, с. 103–107]: дополнение нулями, периодическое продолжение, четное продолжение и др. В главе 2 показывается, что можно обойтись без BCs (и без способов доопределения), и вместо этого излагается новый прием (способ) под названием "усечение–размытие–поворот". Прием "усечение" предложен для того, чтобы не использовать BCs, прием "размытие краев изображения" — для понижения эффекта Гиббса (эффекта ложных волн), а прием "двойной поворот изображения" — для обработки изображения, смазанного под углом по отношению к горизонтали. При

этом методом регуляризации Тихонова решается набор 1-мерных ИУ (в задаче смазывания) или 2-мерное ИУ Фредгольма I рода (в задаче дефокусирования). Решение большого числа примеров показывает, что использование метода регуляризации Тихонова вместе с приемом "усечение–размытие–поворот" заметно повышает точность реконструкции изображений (как серых, так и цветных) по сравнению с методом параметрической фильтрации Винера вместе с приемом ВСс. Применительно к задаче дефокусирования изложен также метод итеративной регуляризации Фридмана.

В главе 3 изложена обратная задача спектроскопии (восстановление истинного спектра по измеренному спектру и аппаратной функции спектрометра). При этом рассматривается случай непрерывного спектра (задача описывается ИУ Фредгольма I рода) и случай дискретного спектра (задача описывается системой линейно-нелинейных уравнений – СЛНУ). Для решения ИУ в случае непрерывного спектра изложен оригинальный способ моделирования, а для решения СЛНУ в случае дискретного спектра изложен оригинальный алгоритм интегральной аппроксимации. В заключении гл. 3 изложены критерий Рэлея и редукционная проблема Рэлея.

Данная книга есть результат многолетней научной работы автора (см. публикации [22, 23, 64, 100, 102, 104, 108–113, 168] и др.), а также педагогической работы в СПбГУ ИТМО и СПбПУ ИМОП (курсы "Обратные прикладные задачи", "Математические основы томографии", "Теория интегральных уравнений", "Прикладная математика", "Теория и технология программирования" и др.).

Автор искренне благодарен А.Л. Агееву, А.С. Апарцину, Д.Г. Арсеньеву, И.В. Бойкову, П.С. Бочарову, В.В. Васину, А.Ф. Верланю, Ю.Е. Воскобойникову, П.А. Галайдину, В.О. Евсееву, В.Ф. Звягину, А.О. Казначеевой, К.А. Кирьянову, А.В. Козаченко, М.А. Койдецки, М.Я. Марусиной, А.В. Меженину, Т.С. Мельниковой, Г.И. Мельникову, В.М. Мусалимову, Ю.И. Неронову, В.Н. Острикову, Ю.П. Петрову, В.В. Пикалову, С.Г. Радченко, Н.Г. Рущенко, С.М. Сарнакову, В.Н. Старкову, В.И. Старостенко, А.Ю. Тропченко, А.А. Фатееву, М.А. Шоморову, Р.А. Экземплярову, А.Г. Яголе за полезные обсуждения вопросов, рассмотренных в книге.

Автор благодарит рецензентов И.П. Гурова и В.М. Иванова, а также эксперта Д.Г. Грязина за труд по изучению рукописи и ценные замечания.

Автор благодарит аспирантов М.В. Дайнеко–Римских и Р.К. Мирджамолова, а также магистранта А.В. Кривых за помощь при наборе книги.

Все замечания и пожелания по данной книге автор просит посылать по адресу: 197183, Санкт-Петербург, ул. Сестрорецкая, 9, кв. 8, В.С. Сизикову или по e-mail: sizikov2000@mail.ru, а также по факсу: (812)2322307 (для Сизикова).

## СПИСОК СОКРАЩЕНИЙ

- BC — boundary condition (граничное условие)
- BCCB — block circulant matrix with circulant blocks  
(блочно-циркулянтная (циклическая) матрица  
с циркулянтными блоками)
- BHNB — BTTB matrix and block Hankel matrix with Hankel blocks  
(BTTB-матрица и блочно-ганкелева матрица  
с ганкелевыми блоками)
- BTTB — block Toeplitz matrix with Toeplitz blocks  
(блочно-теплицева матрица с теплицевыми блоками)
- BTv — bilateral total variation (двусторонняя полная вариация)
- CT — computerized tomography (компьютерная томография)
- CG — conjugate gradients (сопряженные градиенты)
- CMY — Cyan+Magenta+Yellow (голубой+пурпурный+желтый)
- CMYK — Cyan+Magenta+Yellow+black (голубой+пурпурный+желтый+черный)
- FFT — fast Fourier transform (БПФ)
- FID — free induction decay (спад свободной индукции)
- FOV — field of view (поле зрения, область изображения)
- HSI — Hue+Saturation+Intensity  
(цветовой тон+насыщенность+интенсивность)
- IFFT — inverse fast Fourier transform (ОБПФ)
- ind — индексированное изображение
- IPT — пакет Image Processing Toolbox
- map — цветовая карта
- NMR — nuclear magnetic resonance (ЯМР)
- OBC — off-boundary condition (внеграничное условие)
- opt — optimal (оптимальный)
- OTF — optical transfer function (оптическая передаточная функция)
- ppm —  $10^{-6}$  (миллионная доля)
- PSF — Point Spread Function (ФРТ)
- rel — relative (относительный)
- RGB — цветное (полноцветное) изображение
- R,G,B — красный (red), зеленый (green), синий (blue) цвет
- SE — spin echo
- SNR — сигнал/шум
- TE — time echo (время появления эхо-сигнала в ЯМР-томографии)
- TR — repetition time (время повторения серии импульсов)
- TV — total variation (полная вариация)
- uint — unsigned int (беззнаковое целое число)
- АФ — аппаратная функция
- АЦП — аналого-цифровой преобразователь
- АЧХ — амплитудно-частотная характеристика



---

БПФ	— быстрое преобразование Фурье
ВЧ	— высокочастотный (импульс)
ГП	— гауссова плоскость
ДН	— диаграмма направленности (антенны)
ДПФ	— дискретное преобразование Фурье
ИзУ	— измерительное устройство
ИУ	— интегральное уравнение
ИП	— импульсная последовательность
ИП	— индикаторный процесс
КТ	— компьютерная томография
КТХ	— космический телескоп "Хаббл"
МКО	— Международная комиссия по освещению
МНК	— метод наименьших квадратов (Гаусса)
МО	— математическое ожидание
МПОМ	— метод псевдообратной матрицы (Мура–Пенроуза)
МРС	— магнитно-резонансная спектроскопия
МРТ	— магнитно-резонансная томография
НПФ	— непрерывное преобразование Фурье
ОБПФ	— обратное быстрое преобразование Фурье
ОДПФ	— обратное дискретное преобразование Фурье
ОНПФ	— обратное непрерывное преобразование Фурье
опт	— оптимальный
ОПФ	— обратное преобразование Фурье
ОПФ	— оптическая передаточная функция
ПЗС	— прибор с зарядовой связью
ПК	— персональный компьютер
пк, пкс	— пиксель (точка на дискретной 2D-сетке, элемент изображения)
ПФ	— преобразование Фурье
ПЭТ	— позитронно-эмиссионная томография
РКТ	— рентгеновская компьютерная томография
РТ	— рентгеновская томография
СВУ	— специализированное вычислительное устройство
СКО	— среднеквадратическое отклонение
СЛАУ	— система линейных алгебраических уравнений
СЛНУ	— система линейно-нелинейных уравнений
СНУ	— система нелинейных уравнений
СП	— стандартная программа
ССИ	— сигнал спада индукции
СФ	— сканирующая функция
СЧ	— спектральная чувствительность (спектрометра)
Тл	— тесла (единица индукции магнитного поля в системе СИ)
УО	— устройство обработки
УРИ	— устройство регистрации изображений

ФРТ	— функция рассеяния точки
ХН	— характеристика направленности (антенны)
ЦАП	— цифро-аналоговый преобразователь
ЭПР	— электронный парамагнитный резонанс
ЯМР	— ядерный магнитный резонанс

## ВВЕДЕНИЕ. ПРЯМЫЕ И ОБРАТНЫЕ ЗАДАЧИ

Данная книга посвящена изложению некоторых прямых и обратных прикладных задач рентгеновской компьютерной томографии (РКТ), магнитно-резонансной томографии (МРТ), реконструкции искаженных изображений (иконики) и спектроскопии. Дано физическое, техническое и математическое описание задач, изложены некоторые современные, устойчивые методы решения соответствующих уравнений: интегральных уравнений (ИУ), систем линейных алгебраических уравнений (СЛАУ) и т.д., а также приведены тексты разработанных программ в системе MatLab.

*Задачи* бывают *прямые* и *обратные* [102, с. 9, 16].

**О п р е д е л е н и е.** *Прямой задачей* называется формирование выходного процесса на измерительном устройстве (томографе, фотоаппарате, спектрометре, антенне и т.д.) по входному процессу.

**З а м е ч а н и е.** Если прямая задача моделируется на вычислительном устройстве (ПК, СВУ и т.д.), то под формированием выходного процесса будем подразумевать вычисление (расчет) выходного процесса, а если прямая задача выполняется на реальном (натурном) измерительном устройстве, то под формированием выходного процесса будем подразумевать его получение непосредственно измерительным устройством.

### Примеры прямых задач.

1. Пусть дан некоторый *временной процесс*  $y(t)$  (вещественный или комплексный). Одна из прямых задач заключается в вычислении с помощью *преобразования Фурье* спектра процесса ( $\omega$  – круговая частота Фурье):

$$Y(\omega) = \int_{-\infty}^{\infty} y(t) e^{i\omega t} dt, \quad -\infty < \omega < \infty. \quad (1)$$

На рис. 1 приведен пример исходного процесса  $y(t)$  и его преобразования Фурье  $Y(\omega)$  (по модулю). Этот пример в системе программирования MatLab7 (в рамках которой мы будем приводить все примеры) может быть записан следующим образом (ср. [44, с. 438], [45, с. 139]) (см. файл Fig1.m).

```
%Временной процесс в виде трех синусоидальных гармоник и шума:
t = 0:0.0005:1; %2000 шагов по времени t
y = sin(2*pi*200*t) + 0.4*sin(2*pi*150*t) + 0.4*sin(2*pi*250*t);
y1 = y + 0.05*randn(size(t)); %добавление нормального шума [0,0.05]
%График процесса (первые 100 отсчетов):
plot(y1(1:100), '-k', 'LineWidth', 2.5);
grid on; %сетка
%Спектр (ПФ) процесса:
Y1 = fft(y1,1024); %БПФ по первым 1024 отсчетам
M = abs(Y1); %модуль ПФ (АЧХ спектра) 1x1024 double
```

```
f = (0:150)*2000/1024; %первое 151 значение линейной частоты
figure; plot(f,M(1:151),'-k','LineWidth',2.5); %график |Y1|
grid on;
```

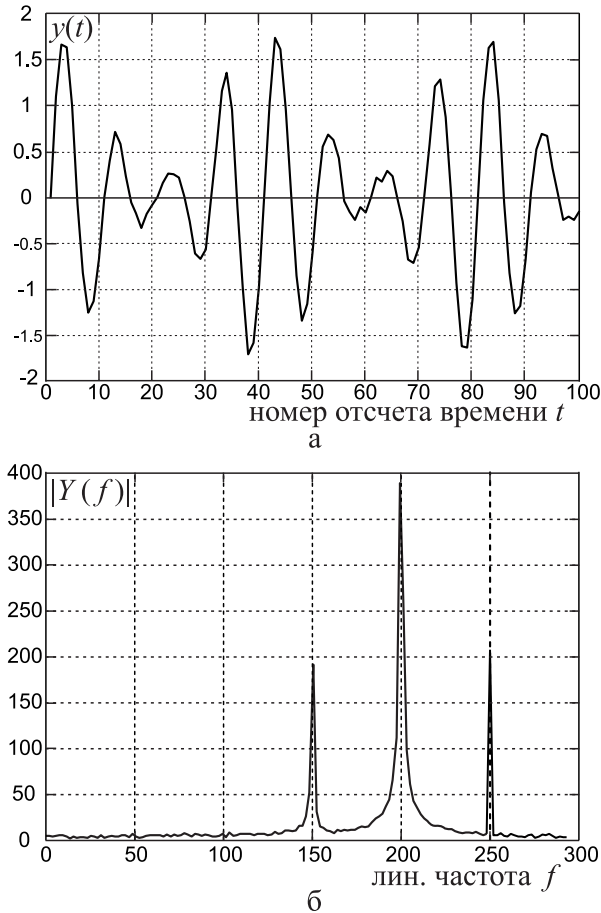


Рис. 1. а — исходный (вещественный) процесс  $y(t)$ , слегка зашумленный;  
б — модуль преобразования Фурье  $|Y(f)|$

2. Пусть задано *распределение плотности*  $c(x, y, z)$  в некотором объекте, а также магнитное поле  $B(x, y, z)$ . Находим (рассчитываем или измеряем) *электромагнитный ЯМР-сигнал*, исходящий из объема  $V$  тела (см. рис. 2) (ср. [102, с. 44, 46]):

$$S(t) = A \iiint_V c(x, y, z) e^{i\gamma B(x, y, z)t} dx dy dz, \quad (2)$$

где  $A$  и  $\gamma$  — некоторые константы (подробности данной задачи ЯМР-томографии см. в гл. 1).

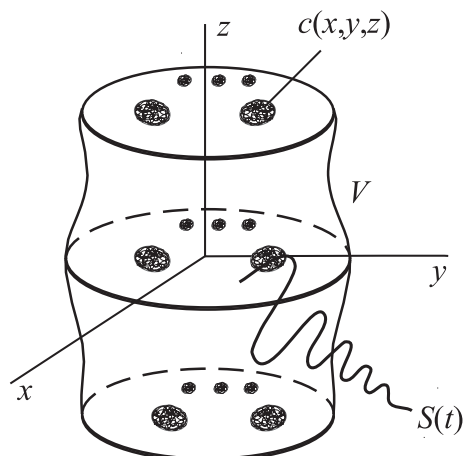


Рис. 2. Объект с рядом  $z$ -сечений в магнитном поле  $B(x, y, z)$  и ЯМР-сигнал  $S(t)$

3. Моделируем *смазывание изображения* согласно формуле [102, с. 66], [120]:

$$g(x, y) = \frac{1}{\Delta} \int_x^{x+\Delta} w(\xi, y) d\xi, \quad (3)$$

где  $w(\xi, y)$  – заданное распределение интенсивности вдоль истинного (неискаженного) изображения,  $g(x, y)$  – распределение интенсивности вдоль смазанного изображения, причем ось  $x$  (и  $\xi$ ) направлена вдоль смаза, а ось  $y$  – перпендикулярно ему,  $\Delta$  – величина смаза (подробности – в гл. 2). Если же смазывание изображения осуществляется "естественным" путем (сдвиг фотоаппарата или движение объекта и т.д.), то функция  $w(\xi, y)$  авторам эксперимента неизвестна (хотя и потенциально существует), а функция  $g(x, y)$  фиксируется (но не рассчитывается) устройством регистрации изображения (фотоаппаратом и т.д.). На рис. 3 в качестве примера приведено истинное изображение, а также смазанное изображение [35, с. 471].

В системе MatLab 7 данный пример был записан следующим образом (см. Fig3.m).

```
clc,clear %очистка Command Window и Workspace
%Чтение изображения:
RGB = imread('girl.jpg'); %256x256x3 uint8
I = rgb2gray(RGB); %преобразование RGB-изображения в серое I
%Вывод изображения (с повышенным контрастом) в 1-е подокно:
figure, subplot(1,2,1), imshow(I,[]), title('I - true (undistorted) image')
%Создание маски (ФРТ, PSF) для смазывания изображения:
PSF = fspecial('motion',30,27); %смаз длиной 30 пкс под углом 27 град.
%Смазывание изображения:
Smearred = imfilter(I,PSF,'circular'); %использование опции 'circular'
%Вывод смазанного изображения во 2-е подокно:
subplot(1,2,2), imshow(Smearred,[]), title('Smearred image')
```



Рис. 3. а — истинное (неискаженное) изображение;  
б — смазанное изображение

4. Моделируем *измерение непрерывного спектра* спектрометром [102, с. 84]:

$$u(\nu) = \int_a^b z(\nu') K(\nu, \nu') d\nu', \quad (4)$$

где  $z(\nu)$  — заданное распределение интенсивности (или энергии) по спектру в функции частоты  $\nu$ ,  $K(\nu, \nu')$  — аппаратная функция (АФ) спектрометра,  $u(\nu)$  — рассчитанное (смоделированное) распределение энергии по измеренному спектру. Если же спектр  $u(\nu)$  получается в результате физического эксперимента, то спектр  $z(\nu)$  неизвестен, аппаратная функция  $K(\nu, \nu')$  известна, а спектр  $u(\nu)$  регистрируется спектрометром. Подробности данной задачи спектроскопии см. в гл. 3. На рис. 4 в качестве примера приведено распределение энергии по истинному спектру  $z(\nu)$  и сглаженное измеренное спектрометром распределение энергии  $u(\nu)$ .

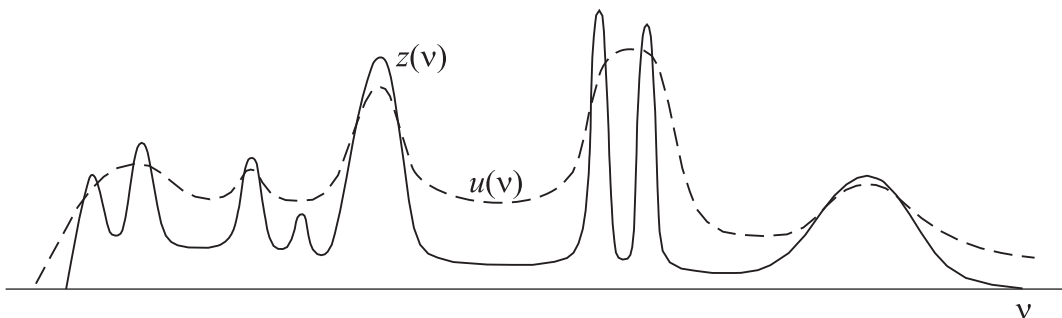


Рис. 4.  $z(\nu)$  — истинный спектр,  $u(\nu)$  — измеренный спектр

В приведенных примерах  $y(t)$ ,  $c(x, y, z)$ ,  $w(x, y)$  и  $z(\nu)$  — входные процессы, а  $Y(\omega)$ ,

$S(t)$ ,  $g(x, y)$  и  $u(\nu)$  – выходные процессы, определяемые в случае моделирования путем вычисления интегралов (ср. [10]).

О п р е д е л е н и е. *Обратной задачей* называется восстановление входного процесса по измеренному выходному процессу.

**Примеры обратных задач.** Это — те же примеры 1–4, но записанные в иной форме.

1. По известному спектру  $Y(\omega)$  нужно *восстановить исходный временной процесс*  $y(t)$ , используя соотношение (1), записанное в виде:

$$\int_{-\infty}^{\infty} y(t) e^{i\omega t} dt = Y(\omega). \quad (1')$$

Соотношение (1') можно рассматривать как интегральное уравнение (ИУ) относительно  $y(t)$ , при этом  $Y(\omega)$  является правой частью, а  $e^{i\omega t}$  – ядром уравнения (подробнее об интегральных уравнениях см. далее). Уравнение (1') есть тот редкий случай, когда его решение находится аналитически:

$$y(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} Y(\omega) e^{-i\omega t} d\omega.$$

2. По измеренному ЯМР-сигналу  $S(t)$  нужно *определить распределение плотности*  $c(x, y, z)$  в некотором объекте. Эта задача сводится к решению интегрального уравнения:

$$A \iiint_V c(x, y, z) e^{i\gamma B(x, y, z)t} dx dy dz = S(t). \quad (2')$$

(подробности — в гл. 1).

3. По измеренному смазанному изображению с распределением интенсивности  $g(x, y)$  нужно *восстановить распределение интенсивности*  $w(x, y)$  вдоль истинного (неискаженного) изображения (восстановить, реконструировать, реставрировать изображение). Задача сводится к решению множества (совокупности) одномерных интегральных уравнений:

$$\frac{1}{\Delta} \int_x^{x+\Delta} w(\xi, y) d\xi = g(x, y) \quad (3')$$

при каждом значении  $y$ , играющем роль параметра (подробности — в гл. 2).

4. Нужно *восстановить истинный спектр*  $z(\nu)$  на основе измеренного спектра  $u(\nu)$  (обратная задача спектроскопии). Задача заключается в решении интегрального уравнения:

$$\int_a^b K(\nu, \nu') z(\nu') d\nu' = u(\nu). \quad (4')$$

Решение уравнения (4') позволит восстановить тонкую структуру спектральной линии, разрешить близкие линии и т.д., другими словами, повысить разрешающую способность спектрометра математическим путем — это один из вариантов редуционной проблемы Рэля (подробности обратной задачи спектроскопии см. в гл. 3).

В приведенных примерах  $Y(\omega)$ ,  $S(t)$ ,  $g(x, y)$  и  $u(\nu)$  — исходные данные, а  $y(t)$ ,  $c(x, y, z)$ ,  $w(\xi, y)$  и  $z(\nu')$  — искомые функции, определяемые путем решения интегральных уравнений (ср. [10, 11]).

Решение обратной задачи обычно значительно сложнее, чем решение прямой задачи. Это аналогично, например, взятию производной  $f'(x)$  от  $f(x)$  в виде формулы, которая практически всегда берется аналитически (будем ее условно рассматривать как прямую задачу), и отысканию первообразной, или неопределенного интеграла  $f(x) = \int f'(x) dx$ , которая обычно берется гораздо сложнее или вообще не берется (будем ее условно рассматривать как обратную задачу).

Обратные задачи обычно формулируются в виде уравнений: систем линейных алгебраических уравнений (СЛАУ), систем нелинейных уравнений (СНУ), систем линейно-нелинейных уравнений (СЛНУ), дифференциальных уравнений или их систем, интегральных уравнений (ИУ) типа (2'), (3'), (4'), а также операторных уравнений типа [102, с. 9, 16]:

$$Ay = f, \quad (5)$$

где  $A$  — некоторый оператор,  $y$  — искомая функция (входной процесс),  $f$  — правая часть (выходной процесс). При этом оператор  $A$  физически и технически ассоциируется с аппаратной функцией (АФ) измерительного устройства, более конкретно, с функцией рассеяния точки (ФРТ) в задаче обработки изображений, спектральной чувствительностью (СЧ) спектрометра в спектроскопии, характеристикой направленности (ХН) антенны в задаче обработки сигналов, импульсной переходной функцией (ИПФ) прибора в общей задаче обработки показаний физических приборов и т.д.

**Две схемы измерений и обработки.** Вышеизложенное можно отобразить в виде следующих *схем* (см. рис. 5 и 6) (ср. [66, с. 17], [102, с. 10]).

Согласно *физико-технической схеме* (рис. 5), на вход измерительного устройства (ИЗУ), например, антенны поступает *входной процесс*  $y$ , равный полезному входному сигналу  $\bar{y}$  и шуму (помехе)  $\delta y$  из внешней среды. Пройдя через ИЗУ, характеризующееся аппаратной функцией (АФ), например, характеристикой направленности (ХН) антенны, сигнал + шум преобразуются в *выходной процесс* — результат измерений, например, сканирующую функцию  $\tilde{f} = f + \delta f$ , где  $f$  — точный (незашумленный) выходной процесс,  $\delta f$  — аппаратная погрешность измерений. Далее следует устройство обработки (УО), цель которого — получить  $\tilde{y}$  (результат обработки), по возможности близкий к входному процессу  $y = \bar{y} + \delta y$  или даже к полезному сигналу  $\bar{y}$ . Следует заметить, что погрешность  $\delta f$  — это мешающий фактор, действие которого нужно стремиться подавить с помощью устройства обработки, а  $\delta y$  в зависимости от некоторого критерия может относиться к помехе, а может относиться и к одной из компонент входного процесса (это характерно для адаптивных методов обработки сигналов — см. [102, п. 3.1]).

Согласно *математической схеме* (рис. 6), оператор измерительного преобразо-



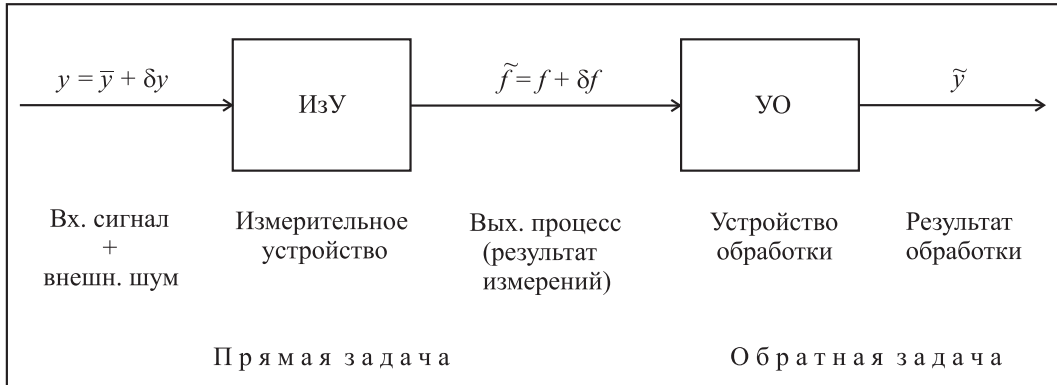


Рис. 5. Физико-техническая схема измерений и обработки

вания  $A$  (соответствующий аппаратной функции) преобразует исходную функцию + внешнюю погрешность  $y = \bar{y} + \delta y$  в правую часть  $\tilde{f} = Ay + \delta f$ , где  $\delta f$  – погрешность правой части. Далее с помощью обратного оператора  $A^{-1}$  вычисляется приближенное решение  $\tilde{y} = A^{-1} \tilde{f}$  и важной задачей является построение такого обратного оператора  $A^{-1}$ , чтобы он давал удовлетворительное приближение к входному процессу  $y$  или даже к исходной функции  $\bar{y}$  и при этом был устойчив по отношению к погрешностям  $\delta f$ . Поскольку многие обратные задачи являются некорректными (см. гл. 2 и 3), то для построения такого обратного оператора обычно используют метод наименьших квадратов Гаусса, метод регуляризации Тихонова, метод фильтрации Винера и т.д., а получаемое решение  $\tilde{y}$  называют псевдорешением, нормальным решением, регуляризованным решением, квазирешением и т.д.

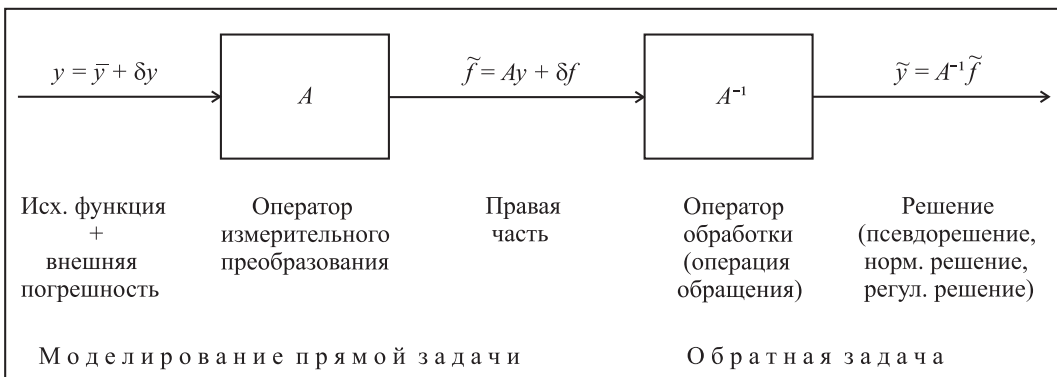


Рис. 6. Математическая схема моделирования измерений и обработки

Решение уравнения (2'), (3'), (4'), (5) и др. позволяет, в принципе, математическим путем восстановить входной процесс и тем самым повысить разрешающую способность измерительного устройства (томографа, фотоаппарата, спектрометра, антенны и т.д.). Сопряжение измерительного устройства с вычислительным устройством (компьютером, спецпроцессором [54,55]) равносильно созданию нового, более совершенного измерительного устройства. Яркий тому пример – реконструкция дефокусирован-

ных изображений космических объектов, полученных на американском космическом телескопе "Хаббл" (КТХ) (см. п. 2.2).

При этом сделаем следующее *з а м е ч а н и е*. Обратные задачи можно разделить на *два типа* (ср. [102, с. 11]).

К задачам *первого типа* отнесем те задачи, в которых искомую функцию можно получить и без математической обработки измерений (за счет совершенствования измерительного устройства). Примеры задач первого типа – это вышеизложенные примеры 3 и 4 (восстановление изображений и восстановление спектра).

К задачам *второго типа* отнесем задачи такого типа, когда никакое совершенствование измерительного устройства не позволит восстановить искомую функцию без математической обработки измерений. Пример задач второго типа – вышеприведенный пример 2 (задача ЯМР-томографии, а также задача рентгеновской компьютерной томографии — см. гл. 1). Роль математических методов при решении задач второго типа гораздо важнее, чем при решении задач первого типа.

### **Контрольные задания и вопросы**

1. Привести примеры прямых прикладных задач.
2. Привести примеры обратных прикладных задач.
3. Объяснить суть прямых и обратных задач на примере уравнения  $Ay = f$ . Что означают  $A$ ,  $y$  и  $f$ , а также что является заданным и что является искомым?
4. Изобразить и объяснить физико-техническую и математическую схемы измерений.
5. Перечислить устойчивые методы обработки измерений.

## Глава 1

# ЗАДАЧИ КОМПЬЮТЕРНОЙ ТОМОГРАФИИ

Слово "томография" происходит от греческих слов *τομή* – сечение и *γραφω* – пишу, т.е. "пишу по сечениям".

Существуют: рентгеновская компьютерная томография — РКТ, ядерно-магнитно-резонансная томография (ЯМР-томография, магнитно-резонансная томография — МРТ), позитронно-эмиссионная томография (ПЭТ), сейсмическая томография, ультразвуковая томография, гравитационная томография и др. [5, 8, 30, 34, 68, 69, 78, 102, 116, 119].

*Суть* всех типов томографии одина: по *суммарной информации*, полученной от некоторого сечения (или слоя) вещества, нужно определить *локальную информацию*, а именно, распределение плотности вещества в сечении  $c(x, y)$ , где  $x, y$  – координаты в сечении, а затем по плотностям  $c_z(x, y)$  в ряде сечений, где  $z$  – координата, перпендикулярная сечению, получить (сконструировать) распределение плотности по объему  $c(x, y, z)$  (см. рис. 1.1).

В разных типах томографии *исходная суммарная информация* качественно различна. В РКТ это интенсивность на детекторах (приемниках), в МРТ это эхо-сигналы и т.д. *Математические описания* также различны: задача РКТ описывается интегральным уравнением Радона или Фредгольма, а в МРТ основным математическим алгоритмом является двухмерное преобразование Фурье и т.д., хотя есть тенденция описать все виды томографии единым так называемым *основным уравнением компьютерной томографии* [119, с. 19, 26, 32, 38]. Однако *конечная цель* одина: получение  $c_z(x, y)$ , поэтому, например, рентгеновские и ЯМР-томограммы (отображение на прозрачных пленках или компьютерном дисплее плотности  $c_z(x, y)$ ) выглядят в принципе одинаково (см. рис. 1.2), хотя физика, техника и математика, необходимые для их получения, различны.

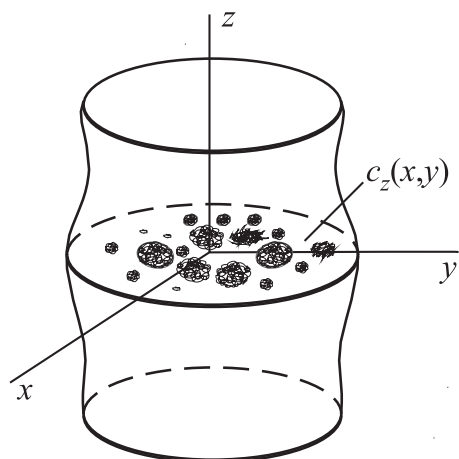


Рис. 1.1. Распределение плотности  $c_z(x, y)$  по  $z$ -сечениям



а



б

Рис. 1.2. а — рентгеновская томограмма,  
б — ЯМР-томограмма

## 1.1. Рентгеновская компьютерная томография (РКТ)

Другие названия РКТ: компьютерная томография (КТ), рентгеновская томография (РТ), X-ray computerized tomography, computerized tomography (СТ), X-ray tomography [21, 69, 78, 102, 116, 119].

Для определенности мы будем излагать основы РКТ применительно к медицине (исследование мозга, руки, грудной клетки и т.д.), хотя область применения РКТ шире (см. дальше).

**Недостатки обычной рентгенографии.** Обычная рентгенография (получение одного рентгеновского снимка, например, грудной клетки) имеет следующие недостатки [116, с. 140]:

1. Различение по плотности соседних участков на рентгеновском снимке возможно лишь в случае, когда их интенсивности (яркости, плотности) отличаются более, чем на  $\approx 2\%$  (если соседние участки отличаются, например на 1%, то они неразличимы по интенсивности, что снижает качество медицинского анализа).

2. Неразличение пространственных структур (если один участок снимаемого объекта заслоняет другой, то последний на рентгеновском снимке не виден).

На рис. 1.3 представлена типичная рентгенограмма.



Рис. 1.3. Типичная рентгенограмма

**Техническая идея РТ.** Недостатки обычной рентгенографии породили идею получения не одного, а нескольких снимков, выполненных под разными углами, и определения по ним плотностей  $c_z(x, y)$  в ряде  $z$ -сечений путем математической обработки снимков.

Конструктивно идея РТ представлена на рис. 1.4, где изображен рентгеновский томограф с параллельным сканированием рентгеновских лучей (ср. [69, с. 10], [78, рис. I.1], [102, с. 20], [116, с. 140]).

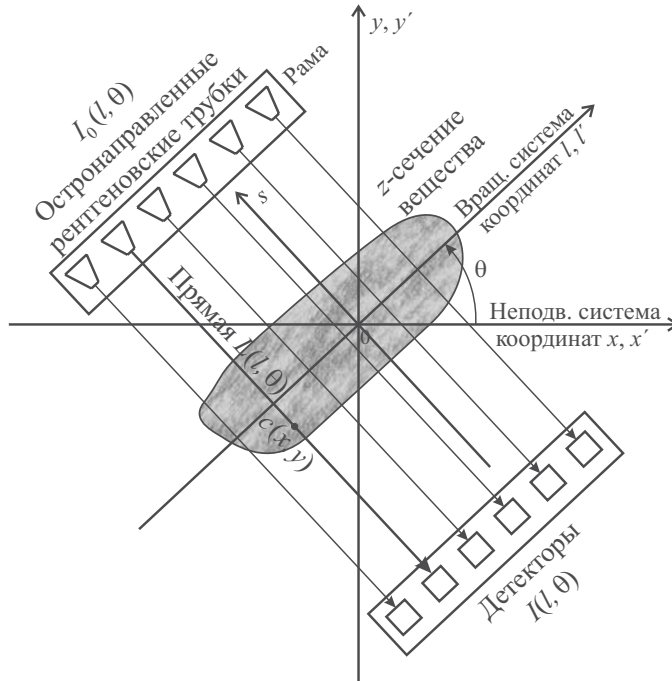


Рис. 1.4. Схема рентгеновского томографа  
(усовершенствованный томограф 1-го поколения)

На раме томографа помещен ряд рентгеновских трубок (источников), излучающих остонаправленные параллельные лучи некоторой (не обязательно одинаковой, но заданной) интенсивности. Эти лучи проходят через сечение (или тонкий слой) объекта, например, мозга и их ослабленные интенсивности фиксируются соответствующими детекторами (приемниками). Затем вся рама с трубками и детекторами поворачивается на новый угол  $\theta$  и эксперимент (излучение и прием) повторяется и т.д. Получив на детекторах двумерную функцию интенсивности  $I(l, \theta)$ , можно путем математической обработки определить двумерную искомую плотность  $c(x, y)$ .

**Пять поколений рентгеновских томографов** [69, с. 10–12], [116, с. 142–146]. Более конкретно, идея РТ реализована в томографах пяти поколений. При этом имеет место параллельное и веерное сканирование рентгеновских лучей [21, с. 19], [116, с. 140, 148].

В томографах *1-го поколения* (1973 г., рис. 1.5) одна остонаправленная рентгеновская трубка и один детектор передвигаются синхронно вдоль рамы для измерения функции  $I(l, \theta_1)$  (1-е сканирование). Затем рама поворачивается на угол  $\theta_2$  и измеряется  $I(l, \theta_2)$  и т.д. Здесь  $l$  – координата трубки и соответствующего детектора (число дискретных значений  $l$  равно  $\approx 160$ ), а  $\theta$  – угол поворота рамы (шаг  $\Delta\theta \approx 1^\circ$ ). Общее время измерений  $T \approx 4$  мин. Если на раме расположить несколько трубок и соответствующих им детекторов (рис. 1.4), то получим усовершенствованный томограф 1-го поколения (пример – первый советский рентгеновский томограф СРТ–1000,

1980 г. [118]).

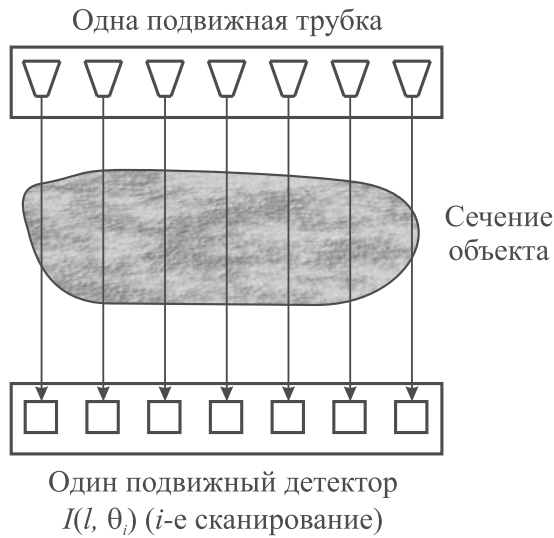
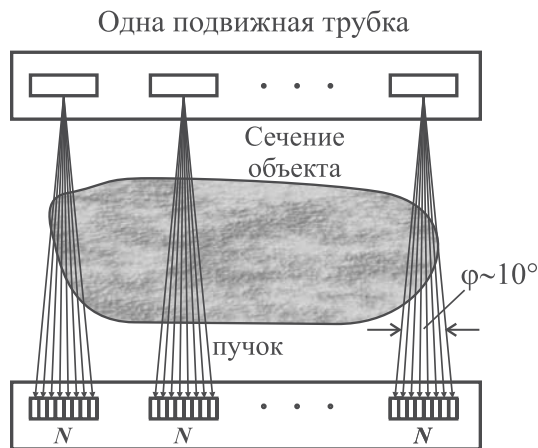


Рис. 1.5. Рентгеновский томограф 1-го поколения

В томографах *2-го поколения*, например, СТ-1010, ЕМІ, Англия (рис. 1.6) одна трубка облучает не один детектор, а  $N \sim 10$  детекторов (объединенных в матрицу детекторов) узким ( $\varphi \sim 10^\circ$ ) веерным рентгеновским пучком.



Одна подвижная матрица из  $N$  детекторов

Рис. 1.6. Рентгеновский томограф 2-го поколения

Затем трубка и матрица детекторов синхронно передвигаются по раме, причем количество положений трубка–матрица детекторов в  $N$  раз меньше, чем в томографах 1-го поколения. При этом  $N$  отдельных лучей в пучке можно считать практически параллельными в силу малости угла  $\varphi$ . Затем рама поворачивается на угол  $\theta$  для следующего сканирования и т.д. Общее время измерений  $T \approx 20$  сек.

Заметим, что в томографах 1-го поколения используется параллельное сканирование рентгеновскими лучами, в томографах 2-го поколения — узковерное сканирование, а начиная с томографов 3-го поколения — (широкое) веерное сканирование.

В томографах *3-го поколения* (рис. 1.7) одна трубка излучает широкий веерный рентгеновский пучок, принимаемый  $\approx 100 - 700$  детекторами, расположенными по дуге. Рама (с трубкой и детекторами) совершает поворот и при каждом ее положении излучается в виде импульса веерный пучок и детекторами измеряется функция (интенсивность)  $I(l, \theta)$ , где  $l$  — криволинейная координата детектора, а  $\theta$  — угол поворота рамы. Время измерений  $T \approx 4 - 5$  сек. Большинство современных медицинских РК-томографов, в том числе, томограф, предложенный в пионерской работе Коренблюма, Тетельбаума и Тютина [56], относятся к томографам 3-го поколения.

В томографах *4-го поколения* (рис. 1.8) (например, в томографе Pfizer 0450, США) имеется сплошное неподвижное кольцо ( $360^\circ$ ) детекторов ( $\sim 1000$ ) и поворачивающаяся рентгеновская трубка. Время измерений  $T \approx 0.1$  сек.

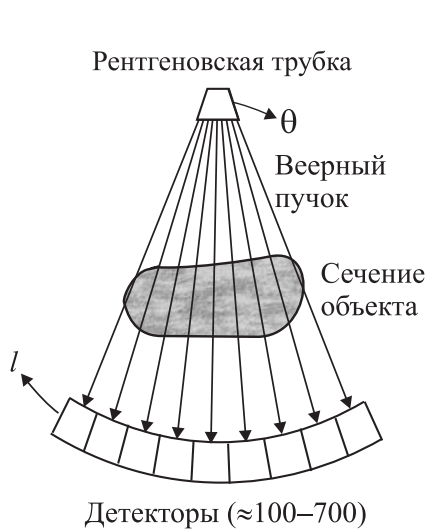


Рис. 1.7. РТ 3-го поколения

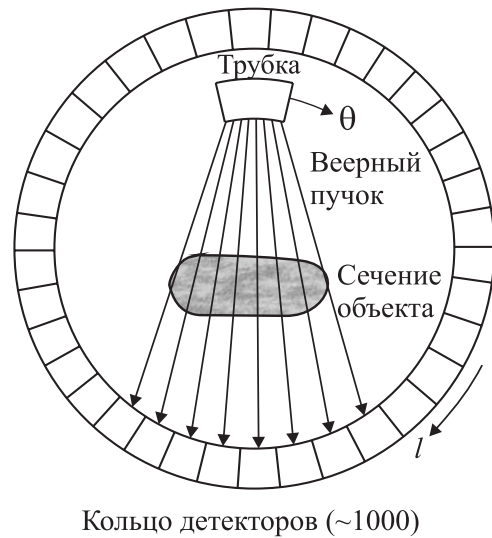


Рис. 1.8. РТ 4-го поколения

В томографах *5-го поколения*, являющихся электронно-лучевыми томографами (начало 1980-х гг., рис. 1.9), поток электронов, создаваемый электронно-лучевой пушкой, направляется на вольфрамовую мишень в виде дуги окружности около  $210^\circ$  и создает рентгеновское излучение в виде рентгеновских пучков. Эти пучки, пройдя через объект (пациента), регистрируются детекторами, расположенными на дуге в  $216^\circ$ . При этом пациент помещается в центр мишени-дуги, а вращение рентгеновских пучков достигается за счет сканирования электронных пучков по мишени. Томографы 5-го поколения не имеют движущихся частей. Время сканирования  $T$  составляет единицы или десятки миллисекунд.

Отметим, что шкала современных рентгеновских компьютерных томографов различает  $\sim 1000$  градаций плотностей выше и ниже плотности воды (от плотности воздуха до плотности кости), а математические методы обработки позволяют различать



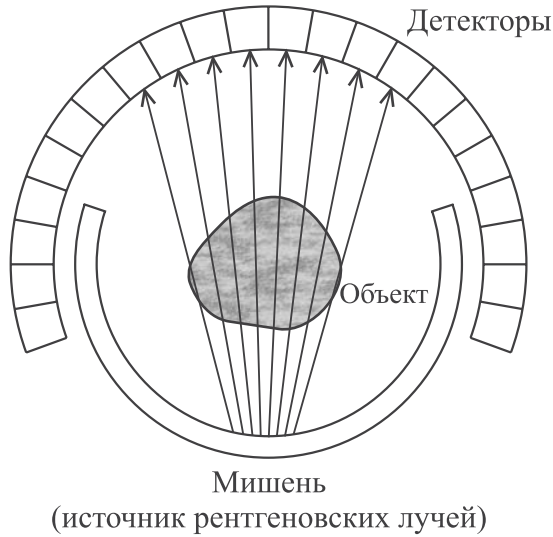


Рис. 1.9. Рентгеновский томограф 5-го поколения

плотности соседних областей до  $\approx 0.5\%$ .

**РКТ-сканер.** Рентгеновский томограф вместе с дополнительными устройствами (компьютер, ложе пациента, кожух и т.д.) образуют так называемый *РКТ-сканер* (рис. 1.10) [70], [116, с. 142]:



Рис. 1.10. Типичный РКТ-сканер

В нем выполняется "просвечивание" пациента рентгеновскими лучами, математическая обработка результатов "просвечивания" и визуализация результатов обработки (вывод томограммы на монитор).

Теперь рассмотрим самый важный вопрос — математическую обработку результатов измерений рентгеновских лучей, прошедших через объект (пациента). Рассмотрим этот вопрос на примере работы усовершенствованного томографа 1-го поколения

(рис. 1.4), т.е. томографа с параллельным сканированием рентгеновских лучей, хотя нижеследующие математические соотношения будут справедливы в принципе и для томографов с веерным сканированием, нужно только учитывать, что  $l$  является криволинейной координатой (см. рис. 1.7).

**Закон Бера.** Введем неподвижную относительно объекта систему координат  $x, y$  и вращающуюся, связанную с рамой, систему координат  $l, s$ . Тогда согласно закону Бера (точнее, закону Бугера–Ламберта–Бера, или обобщенному закону Бера) [116, с. 147], [128, с. 60], интенсивность рентгеновского луча, принятого некоторым детектором, равна

$$I(l, \theta) = I_0(l, \theta) \cdot \exp \left[ - \int_{L(l, \theta)} c(x, y) ds \right], \quad (1.1)$$

где  $l$  – координата детектора (криволинейная в случае томографов 3–5 поколений),  $\theta$  – угол поворота рамы,  $I_0(l, \theta)$  – интенсивность соответствующей излучающей трубки (обычно  $I_0(l, \theta) = \text{const} = I_0$ ),  $c(x, y)$  – плотность вещества (точнее, коэффициент поглощения рентгеновских лучей веществом) на луче  $L(l, \theta)$ , являющемся прямой, уравнение которой:

$$x \cos \theta + y \sin \theta = l. \quad (1.2)$$

Интегрирование в (1.1) ведется по лучу  $L(l, \theta)$ . Интеграл  $\int_{L(l, \theta)} c(x, y) ds$  есть *масса вещества на луче*. Чем больше значение этого интеграла, тем меньше принятая интенсивность  $I(l, \theta)$ . Как будет видно из дальнейшего изложения, выражение (1.1) лежит в основе математического аппарата РКТ.

**Преобразование и уравнение Радона.** Запишем (1.1) иным образом (полагая  $I_0(l, \theta) = \text{const} = I_0$ ):

$$I(l, \theta)/I_0 = e^{-q(l, \theta)}, \quad (1.3)$$

где

$$q(l, \theta) = \int_{L(l, \theta)} c(x, y) ds. \quad (1.4)$$

**О п р е д е л е н и е.** Выражение (1.4), где  $L(l, \theta)$  есть некоторый луч, проходящий через вещество,  $c(x, y)$  – плотность вещества на этом луче, а координата  $s$  направлена вдоль луча, называется *преобразованием Радона* (радоновским образом, теньевым преобразованием) (1917 г.) [21, 69, 78, 102, 116, 119, 123]. Функцию  $q(l, \theta)$  обозначают также через  $R(l, \theta)$ .

Логарифмируя (1.3), получим:

$$q(l, \theta) = - \ln [I(l, \theta)/I_0]. \quad (1.5)$$

**О п р е д е л е н и е.** Функция  $q(l, \theta)$  называется *поглощением* (тенью изображения, исходной интегральной проекцией).

**О п р е д е л е н и е.** Отношение  $I(l, \theta)/I_0$  называется *прозрачностью*.

**Ч а с т н ы е с л у ч а и.** Если  $c(x, y) = 0$  (вакуум), то  $I(l, \theta)/I_0 = 1$  (полная прозрачность среды), а  $q(l, \theta) = 0$  (поглощения нет).

Если же  $c(x, y) = \infty$  (абсолютно непрозрачная среда), то  $I(l, \theta)/I_0 = 0$  (нулевая прозрачность), а  $q(l, \theta) = \infty$  (бесконечно большое поглощение).

Запишем (1.4) иначе (поменяем местами левую и правую части этого соотношения):

$$\int_{L(l, \theta)} c(x, y) ds = q(l, \theta). \quad (1.6)$$

Соотношение (1.6) есть *уравнение Радона* (формула Радона, уравнение тени) [69, 78, 102, 116, 119, 123]. Оно является основным уравнением РТ. В нем двухмерная функция  $q(l, \theta)$  определяется согласно (1.5), где  $I(l, \theta)$  получается в результате измерений. А двухмерная функция  $c(x, y)$  является искомой функцией. Поэтому (1.6) можно рассматривать как *интегральное уравнение* относительно  $c(x, y)$  на основе измеренной правой части  $q(l, \theta)$ .

Решение уравнения (1.6) позволяет в принципе найти распределение плотности  $c(x, y)$  в некотором сечении объекта (например, мозга) по измеренной  $I(l, \theta)$ , а значит, и  $q(l, \theta)$ . Такая задача называется *реконструкцией рентгеновского изображения*.

Впервые задачу реконструкции рентгеновского изображения рассмотрел Радон в 1917 г. [116, с. 154], [169]. Однако уравнение (1.6) имеет следующие особенности, затрудняющие его решение.

Во-первых, уравнение (1.6) как интегральное уравнение имеет нестандартную форму, а именно, в нем нет в явном виде ядра; хотя правая часть  $q(l, \theta)$  и искомая функция  $c(x, y)$  являются двухмерными, интеграл является одномерным; интеграл не имеет в явном виде нижнего и верхнего пределов интегрирования.

Во-вторых, задача решения уравнения (1.6) является *некорректной* (сильно *неустойчивой* и т.д.).

В-третьих, хотя уже Радон получил одно из решений уравнения (1.6) [119, с. 40]:

$$c(x, y) = -\frac{1}{2\pi^2} \int_0^\pi d\theta \int_{-\infty}^\infty \frac{\partial q(l, \theta)}{\partial l} \frac{dl}{l - (x \cos \theta + y \sin \theta)}, \quad (1.7)$$

но это решение является неустойчивым из-за необходимости численного вычисления производной  $\partial q(l, \theta)/\partial l$ , кроме того, второй интеграл в (1.7) является сингулярным, так как знаменатель  $l - (x \cos \theta + y \sin \theta)$  может обращаться в нуль (аналогичное решение было получено в работе [56]).

Поэтому задача решения уравнения (1.6) в течение примерно 4–5 десятилетий после пионерских работ Радона практически не рассматривалась математиками. И лишь с 60-х гг. к ней снова вернулись, когда были разработаны методы решения некорректных задач, нестандартных и сингулярных интегральных уравнений и т.д.

Рассмотрим некоторые современные эффективные

**Методы решения уравнения Радона (1.6).**

В ряде методов используется

**Теорема Брейсуэлла о сечении спектра** [146] (другое название: теорема о центральном, или проекционном слое [123]). Она связывает спектры (преобразования

Фурье) искомой функции  $c(x, y)$  и правой части  $q(l, \theta)$ :

$$\hat{c}(\omega \cos \theta, \omega \sin \theta) = \hat{q}(\omega, \theta), \quad (1.8)$$

где

$$\hat{c}(\omega_1, \omega_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c(x, y) e^{i(\omega_1 x + \omega_2 y)} dx dy \quad (1.9)$$

— двумерное преобразование Фурье (ПФ) от  $c(x, y)$ ,

$$\hat{q}(\omega, \theta) = \int_{-\infty}^{\infty} q(l, \theta) e^{i\omega l} dl \quad (1.10)$$

— одномерное ПФ от  $q(l, \theta)$  по переменной  $l$  при фиксированном  $\theta$ .

Доказательство. Запишем (1.4) в виде:

$$q(l, \theta) = \int_{-\infty}^{\infty} c(x, y) ds.$$

Выполним одномерное ПФ по переменной  $l$  при фиксированном  $\theta$ :

$$\hat{q}(\omega, \theta) = \int_{-\infty}^{\infty} q(l, \theta) e^{i\omega l} dl = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c(x, y) e^{i\omega l} ds dl.$$

Поскольку  $ds dl = dx dy$  и учитывая (1.2), получим:

$$\begin{aligned} \hat{q}(\omega, \theta) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c(x, y) e^{i\omega(x \cos \theta + y \sin \theta)} dx dy = \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c(x, y) e^{i(\omega \cos \theta \cdot x + \omega \sin \theta \cdot y)} dx dy = \hat{c}(\omega \cos \theta, \omega \sin \theta). \end{aligned}$$

Теорема доказана. Эта теорема носит название "о сечении спектра" потому, что фигурирующие в ней спектры (преобразования Фурье)  $\hat{c}$  и  $\hat{q}$  находятся для ряда фиксированных значений  $\theta$ , т.е. выполняется ряд сечений спектров.

Теперь рассмотрим два метода решения уравнения Радона (1.6) на основе теоремы Брейсуэлла.

**Метод преобразования Фурье** [116, 123]. Из (1.9) можно получить искомое решение в виде двумерного обратного преобразования Фурье (ОПФ):

$$c(x, y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \hat{c}(\omega_1, \omega_2) e^{-i(\omega_1 x + \omega_2 y)} d\omega_1 d\omega_2. \quad (1.11)$$

Однако непосредственно функция  $\hat{c}(\omega_1, \omega_2)$  неизвестна, а известна функция  $\hat{c}(\omega \cos \theta, \omega \sin \theta)$  через соотношение (1.8) (и (1.10)). Чтобы найти  $\hat{c}(\omega_1, \omega_2)$ , обозначим:

$$\omega \cos \theta = \omega_1, \quad \omega \sin \theta = \omega_2,$$

откуда

$$\omega = \sqrt{\omega_1^2 + \omega_2^2}, \quad \theta = \arctg(\omega_2/\omega_1).$$

В результате соотношение (1.8) можно записать в виде:

$$\hat{c}(\omega_1, \omega_2) = \hat{q}\left(\sqrt{\omega_1^2 + \omega_2^2}, \arctg(\omega_2/\omega_1)\right). \quad (1.12)$$

Практически метод ПФ выглядит следующим образом.

1) Вычисляем спектр  $\hat{q}(\omega, \theta)$  согласно (1.10) на некоторых дискретных сетках узлов  $\omega$  и  $\theta$ .

2) Задаем сетки узлов по  $\omega_1$  и  $\omega_2$  и пересчитываем  $\hat{q}(\omega, \theta)$  на эти сетки, а именно, используя интерполяцию, рассчитываем  $\hat{q}(\sqrt{\omega_1^2 + \omega_2^2}, \arctg(\omega_2/\omega_1))$ .

3) Получаем  $\hat{c}(\omega_1, \omega_2)$  согласно (1.12).

4) Вычисляем искомую функцию  $c(x, y)$  согласно (1.11).

**Метод свертки и обратной проекции (три варианта)** [78, 116, 123]. Как и в предыдущем методе, запишем решение  $c(x, y)$  в виде ОПФ (1.11). Переходя к полярным координатам  $(\omega, \theta)$  так, что  $\omega_1 = \omega \cos \theta$ ,  $\omega_2 = \omega \sin \theta$ ,  $d\omega_1 d\omega_2 = \omega d\omega d\theta$ , получим:

$$c(x, y) = \frac{1}{4\pi^2} \int_0^{2\pi} d\theta \int_0^{\infty} \omega \hat{c}(\omega \cos \theta, \omega \sin \theta) e^{-i\omega(x \cos \theta + y \sin \theta)} d\omega.$$

Далее, учитывая (1.8) и (1.10), получим:

$$c(x, y) = \frac{1}{4\pi^2} \int_0^{\pi} d\theta \int_{-\infty}^{\infty} q(l, \theta) \left[ \int_{-\infty}^{\infty} |\omega| e^{-i\omega(x \cos \theta + y \sin \theta - l)} d\omega \right] dl$$

или

$$c(x, y) = \frac{1}{4\pi^2} \int_0^{\pi} d\theta \int_{-\infty}^{\infty} q(l, \theta) p(x \cos \theta + y \sin \theta - l) dl, \quad (1.13)$$

где

$$p(t) = \int_{-\infty}^{\infty} |\omega| e^{-i\omega t} d\omega \quad (1.14)$$

— так называемая *импульсная реакция фильтра* с частотной характеристикой  $|\omega|$  [123].

Исследуем кратко решение (1.13)–(1.14). Воспользуемся формулой Эйлера  $e^{-i\omega t} = \cos \omega t - i \sin \omega t$  и тем, что  $\int_{-\infty}^{\infty} |\omega| \cos \omega t d\omega = 2 \int_0^{\infty} \omega \cos \omega t d\omega$  как интеграл от четной функции и  $\int_{-\infty}^{\infty} |\omega| \sin \omega t d\omega = 0$  как интеграл от нечетной функции [72, с. 6]. Получим:

$$p(t) = 2 \int_0^{\infty} \omega \cos \omega t d\omega. \quad (1.15)$$

Однако подынтегральная функция в (1.15) (обозначим ее  $v(\omega) = \omega \cos \omega t$ ) является расходящейся функцией при любом  $t \in (-\infty, \infty)$  (назовем ее расходящимся косинусом) (см. рис. 1.11).

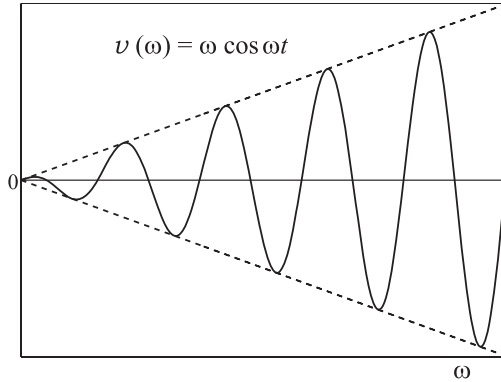


Рис. 1.11. Подынтегральная функция  $v(\omega) = \omega \cos \omega t$  при некотором  $t$  (пунктирные линии — огибающие)

Поэтому интеграл в (1.15) не имеет сходимости при любом  $t \in (-\infty, \infty)$  и, как следствие, не имеет сходимости и выражение (1.13) при любых  $x, y \in (-\infty, \infty)$ . Другими словами, решение (1.13)–(1.15) есть расходящееся решение даже при точной  $q(l, \theta)$ .

Рассмотрим три варианта "корректировки" функции  $p(t)$  так, чтобы интеграл в (1.15) сходилась и выражение (1.13) также сходилась и было устойчивым по отношению к погрешностям  $q(l, \theta)$ .

В *первом варианте* (Рамачандран и Лакшминараян, 1971) [78, с. 126], [116, с. 156, 171], [170] учитывается, что на практике максимальная частота  $\omega_{\max} \neq \infty$ . Если шаг дискретизации  $h = \Delta x = \Delta y = \Delta l = \Delta t = \text{const}$ , то в соответствии с теоремой Котельникова,  $\omega_{\max} = \pi/h$  (частота Найквиста). Поэтому

$$\begin{aligned} p_R(t) &= 2 \int_0^{\omega_{\max}} \omega \cos \omega t d\omega = 2 \left[ \frac{\cos(\omega_{\max} t) - 1}{t^2} + \omega_{\max}^2 \frac{\sin(\omega_{\max} t)}{\omega_{\max} t} \right] = \\ &= -\frac{\sin^2(\omega_{\max} t/2)}{(t/2)^2} + 2\omega_{\max}^2 \frac{\sin(\omega_{\max} t)}{\omega_{\max} t} = \omega_{\max}^2 \left[ 2 \frac{\sin(\omega_{\max} t)}{\omega_{\max} t} - \frac{\sin^2(\omega_{\max} t/2)}{(\omega_{\max} t/2)^2} \right] \end{aligned} \quad (1.16)$$

или

$$p_R(t) = \omega_{\max}^2 [2 \text{sinc}(\omega_{\max} t) - \text{sinc}^2(\omega_{\max} t/2)], \quad (1.17)$$

где  $\text{sinc}(\omega_{\max}t) = \sin(\omega_{\max}t)/(\omega_{\max}t)$ . На рис. 1.12 представлена непрерывная кривая (1.17) (*фильтр Рамачандрана–Лакшиминарайяна* [78, 116, 170]), причем для удобства по горизонтальной оси отмечено не  $t$ , а  $\omega_{\max}t$ , а по вертикальной оси —  $p_R(t)/\omega_{\max}^2$ . Кружками показаны дискретные отсчеты этой кривой при  $t = 0, \pm h, \pm 2h, \dots$ , т.е. при  $\omega_{\max}t = 0, \pm\pi, \pm 2\pi, \dots$

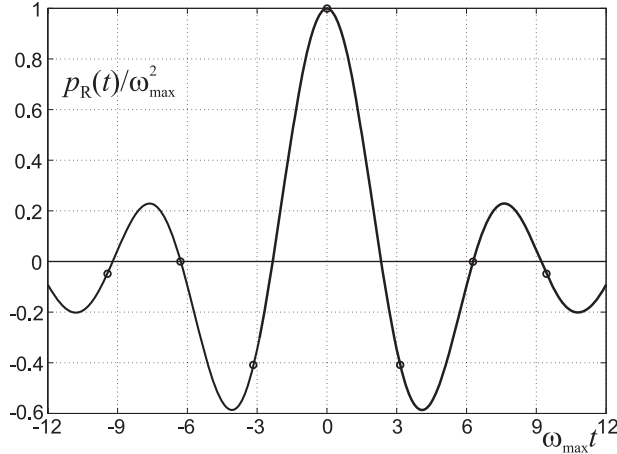


Рис. 1.12. Зависимость  $p_R(t)/\omega_{\max}^2$  от  $\omega_{\max}t$

Недостатками фильтра (1.17) являются резкое усечение высоких частот и высокая чувствительность фильтра к частотам вблизи  $\omega_{\max}$ . Во *втором варианте* (фильтр Шеппа–Логана) [78, с. 127], [116, с. 161], [176] выполнено более умеренное подавление высоких Фурье-частот за счет введения множителя  $\text{sinc}(\pi\omega/\omega_{\max})$  в подынтегральное выражение в (1.16). В результате

$$p_S(t) = 2 \int_0^{\omega_{\max}} \omega \text{sinc}(\pi\omega/\omega_{\max}) \cos \omega t \, d\omega = \omega_{\max}^2 4 \frac{\cos^2(\omega_{\max}t/2)}{\pi^2 - (\omega_{\max}t)^2}. \quad (1.18)$$

Это равносильно использованию известного метода  $\sigma$ -множителей Ланцоша [22, с. 259–260]. На рис. 1.13 представлена кривая (1.18) — более гладкая, чем кривая  $p_R(t)$ .

Еще более аккуратное подавление высоких частот Фурье  $\omega$  в выражении (1.14) для фильтра  $p(t)$  выполняется с помощью метода регуляризации Тихонова (*третий вариант*, модифицированный фильтр Троицкого) [101, 123, 175]:

$$p_\alpha(t) = 2 \int_0^{\infty} \frac{\omega}{1 + \alpha \omega^{2p}} \cos \omega t \, d\omega, \quad (1.19)$$

где  $\alpha > 0$  — параметр регуляризации,  $p = 1, 2, 3, \dots$  — порядок регуляризации. Обычно  $p = 1$  или 2, а для выбора значения  $\alpha$  существует ряд способов (обобщенный принцип невязки, способ подбора и др. [22, с. 242–249]). Обычно  $\alpha \sim 10^{-3} \div 10^{-6}$ , если погрешность  $q(l, \theta)$  порядка 0.1 — 1%. На рис. 1.14а приведена подынтегральная функция

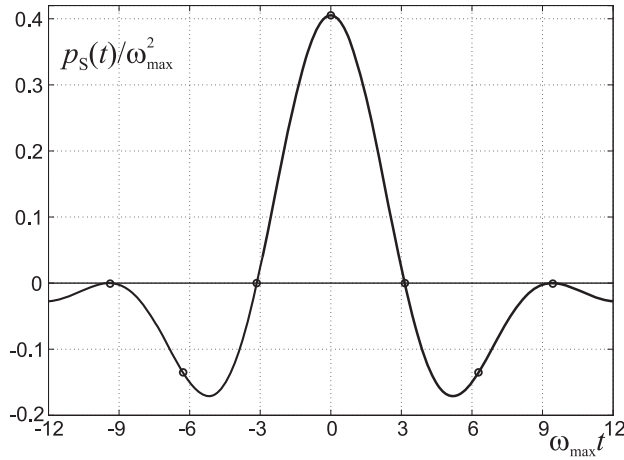


Рис. 1.13. Зависимость  $p_S(t)/\omega_{\max}^2$  от  $\omega_{\max}t$

$v_\alpha(\omega) = \omega/(1 + \alpha\omega^{2p}) \cdot \cos(\omega t)$  при  $t = 1$ ,  $\alpha = 10^{-4}$ ,  $p = 2$ . Интеграл (1.19) от нее сходится, причем с умеренным подавлением высоких частот Фурье  $\omega$ . На рис. 1.14б приведена функция  $p_\alpha(t)$ .

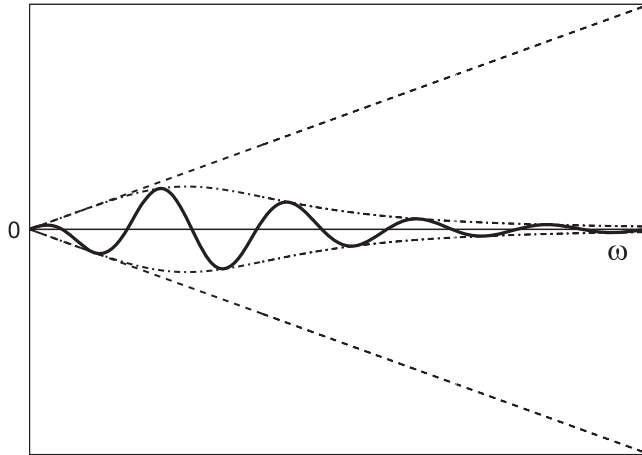
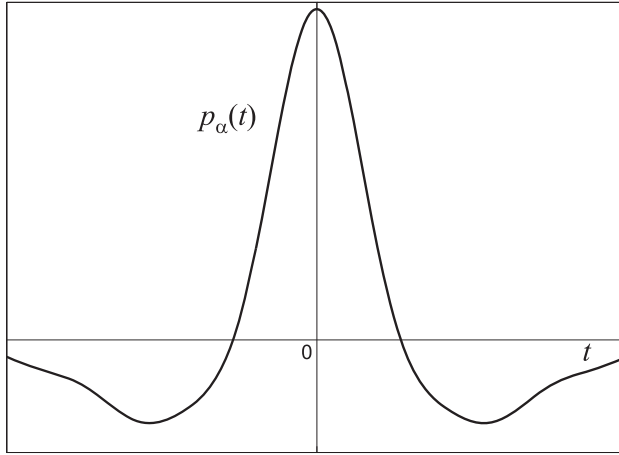


Рис. 1.14а. Подынтегральная функция  $v_\alpha(\omega)$  (непрерывная линия) при  $t = 1$ ,  $\alpha = 10^{-4}$ ,  $p = 2$ ;  
 пунктир — грубая огибающая  $\pm\omega$ ;  
 штрих-пунктир — близкая огибающая  $\pm\omega/(1 + \alpha\omega^{2p})$

**Приведение уравнения Радона к уравнению Фредгольма I рода.** Интегральное уравнение Радона (1.6), как мы уже отметили, имеет нестандартную форму (отсутствует ядро; интеграл является одномерным, хотя функции  $s(x, y)$  и  $q(l, \theta)$  являются двухмерными; нет пределов интегрирования) и это затрудняет его решение. В работе [119, с. 65] уравнение (1.6) приведено к стандартной форме — *двухмерному*




 Рис. 1.14в. Функция  $p_\alpha(t)$  (см. (1.19))

интегральному уравнению Фредгольма I рода типа свертки [102, с. 21]:

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{c(x', y') dx' dy'}{\sqrt{(x-x')^2 + (y-y')^2}} = S(x, y), \quad (1.20)$$

где

$$S(x, y) = \frac{1}{\pi} \int_0^\pi q(x \cos \theta + y \sin \theta, \theta) d\theta. \quad (1.21)$$

Уравнение (1.20) имеет стандартную форму. В нем (разностное) ядро равно  $K(x-x', y-y') = 1/\sqrt{(x-x')^2 + (y-y')^2}$  или

$$K(x, y) = \frac{1}{\sqrt{x^2 + y^2}}, \quad (1.22)$$

искомая функция — это  $c(x, y)$ , а правая часть  $S(x, y)$  может быть вычислена численно по известной  $q(t, \theta)$  согласно (1.21).

Методы решения уравнения вида (1.20) хорошо разработаны [22, с. 268]. Основной метод решения уравнения (1.20) — это метод двумерного преобразования Фурье (ПФ) [22, с. 268–269], [102, с. 22]. При этом нужно учитывать, что задача решения уравнения (1.20) является *некорректной*. Поэтому необходимо добавить к методу ПФ еще некоторый устойчивый метод, в первую очередь, метод регуляризации Тихонова. Отметим еще, что уравнение (1.20) является *сингулярным* (его ядро при  $x = x'$  и  $y = y'$  обращается в бесконечность). Однако, как видно далее, преобразование Фурье от ядра (1.22) автоматически устраняет сингулярность.

**Решение уравнения Фредгольма методом ПФ с регуляризацией Тихонова.** В п. 2.3 и 2.5 изложен метод ПФ решения двумерного интегрального уравнения Фредгольма I рода типа свертки. Согласно нему, решение уравнения (1.20) имеет

вид [102, с. 22]:

$$c(x, y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \hat{c}(\omega_1, \omega_2) e^{-i(\omega_1 x + \omega_2 y)} d\omega_1 d\omega_2, \quad (1.23)$$

где ПФ (спектр) решения равно

$$\hat{c}(\omega_1, \omega_2) = \frac{\hat{S}(\omega_1, \omega_2)}{\hat{K}(\omega_1, \omega_2)} = \frac{\sqrt{\omega_1^2 + \omega_2^2}}{2\pi} \hat{S}(\omega_1, \omega_2), \quad (1.24)$$

причем

$$\hat{S}(\omega_1, \omega_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} S(x, y) e^{i(\omega_1 x + \omega_2 y)} dx dy$$

— ПФ правой части (вычисляется численно),

$$\hat{K}(\omega_1, \omega_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} K(x, y) e^{i(\omega_1 x + \omega_2 y)} dx dy = \frac{2\pi}{\sqrt{\omega_1^2 + \omega_2^2}} = \frac{2\pi}{\omega}$$

— ПФ ядра (вычисляется аналитически),  $\omega = \sqrt{\omega_1^2 + \omega_2^2}$ .

Однако решение (1.23) неустойчиво (проявление некорректности).

В *методе регуляризации Тихонова* вместо (1.23) и (1.24) используются формулы [22, с. 269], [102, с. 22], [119, с. 68]:

$$c_\alpha(x, y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \hat{c}_\alpha(\omega_1, \omega_2) e^{-i(\omega_1 x + \omega_2 y)} d\omega_1 d\omega_2 \quad (1.25)$$

— регуляризованное решение,

$$\begin{aligned} \hat{c}_\alpha(\omega_1, \omega_2) &= \frac{\hat{K}(-\omega_1, -\omega_2) \hat{S}(\omega_1, \omega_2)}{|\hat{K}(\omega_1, \omega_2)|^2 + \alpha \omega^{2p}} = \\ &= \frac{2\pi}{\omega} \frac{\hat{S}(\omega_1, \omega_2)}{[(\frac{2\pi}{\omega})^2 + \alpha \omega^{2p}]} = \frac{\omega}{2\pi} \frac{\hat{S}(\omega_1, \omega_2)}{[1 + \frac{\alpha}{(2\pi)^2} \omega^{2(p+1)}]} \end{aligned} \quad (1.26)$$

— регуляризованный спектр решения, где  $\alpha > 0$  — параметр регуляризации,  $p$  — порядок регуляризации (обычно  $p = 1$  или  $2$ ). Если для упрощения записи заменить  $\alpha/(2\pi)^2$  на  $\alpha$ , а  $p + 1$  на  $p$ , то (1.26) запишется в виде:

$$\hat{c}_\alpha(\omega_1, \omega_2) = \frac{1}{2\pi} \frac{\omega}{1 + \alpha \omega^{2p}} \hat{S}(\omega_1, \omega_2), \quad (1.27)$$

а, например, в работе [119, с. 68] регуляризованный спектр записывается в виде:

$$\hat{c}_\alpha(\omega_1, \omega_2) = \frac{1}{2\pi} \frac{\omega}{1 + \alpha \omega^2 (1 + \omega^4)} \hat{S}(\omega_1, \omega_2). \quad (1.28)$$

Неустойчивость решения (1.23) можно объяснить следующим образом. Интенсивности на детекторах  $I(l, \theta)$  измеряются с погрешностями. Поэтому функции  $q(l, \theta)$  (см. (1.5)) и  $S(x, y)$  (см. (1.21)) будут также иметь погрешности. При этом погрешности обычно имеют компоненту "белого шума" — постоянную (не зависящую от частот  $\omega_1$  и  $\omega_2$ ) компоненту. В результате  $\hat{S}(\omega_1, \omega_2) \rightarrow \text{const}$  при  $\omega_1, \omega_2 \rightarrow \infty$ , а  $\hat{c}(\omega_1, \omega_2) \rightarrow \infty$  при  $\omega_1, \omega_2 \rightarrow \infty$  (см. (1.24)) и интеграл в (1.23) расходится. На практике интеграл (1.23) как ОНПФ вычисляется дискретно (с помощью ДПФ или БПФ) до конечных максимальных значений частот  $\omega_1$  и  $\omega_2$ , т.е. имеет место усечение Фурье-частот и эффект неустойчивости понижается, хотя и остается.

В методе регуляризации Тихонова, во-первых,  $\hat{c}_\alpha(\omega_1, \omega_2) \rightarrow 0$  при  $\omega \rightarrow \infty$  (см. (1.27) при  $p > 0.5$  и (1.28)) и интеграл (1.25) сходится. Во-вторых, подавление высоких Фурье-частот  $\omega$  в (1.27) и в (1.28) происходит более аккуратно, чем в случае усечения по частоте. Дело в том, что с одной стороны, высокие частоты наиболее сильно реагируют на погрешности и поэтому их нужно подавлять, а с другой стороны, высокие частоты необходимы для повышения разрешения томограмм (выделение близких и/или мелких деталей и т.д.). Поэтому это подавление должно быть умеренным. Обоим этим критериям (в пропорции, зависящей от параметра регуляризации  $\alpha$ ) удовлетворяет метод регуляризации Тихонова.

**Историческая справка** [69, с. 9–10], [116, с. 154]. Впервые задачу реконструкции рентгеновского изображения рассмотрел Радон (Австрия) в 1917 г. [169]. Он же получил решение уравнения (1.6) в виде (1.7). Однако из-за некорректности (неустойчивости) этого решения и технических проблем эта задача в течение примерно четырех десятилетий находилась в тени. И лишь в 1956–58 гг. она была вновь переоткрыта Тетельбаумом, Коренблюмом и Тютиным (СССР), разработавшими первую систему реконструкции рентгеновских медицинских изображений [56], [116, с. 154].

В 1961 г. Олдендорф (США) предложил метод компьютерной томографии. В 1963 г. Кормак (США) провел эксперименты по рентгеновской томографии. В 1972 г. была получена первая достаточно качественная томограмма головного мозга человека.

В 1967 г. Брейсуэлл и Риддл (Австрия–США) [146] доказали знаменитую теорему о сечении спектра, на основе которой были разработаны метод преобразования Фурье, метод свертки и обратной проекции, а также фильтр Рамачандрана–Лакшминараяна (Индия–США, 1971 г.) [170], фильтр Шеппа–Логана (Англия, 1974 г.) [176] и фильтр Троицкого на основе метода регуляризации Тихонова (СССР, 1980-е гг.) [123].

В 1973 г. Хаунсфилд (Англия) разработал первую на Западе коммерческую систему — сканер головного мозга английской фирмы ЕМІ [156]. В 1978 г. был разработан первый отечественный медицинский рентгеновский томограф СРТ–1000 (руководитель Рубашов, ЦНИИ КТ), в котором заложено решение уравнения (1.20) методом регуляризации [118].

В 1979 г. Хаунсфилд и Кормак, а в 1982 г. Клуг (Англия) (за применение РТ в биохимии) получили Нобелевские премии.

**Численные иллюстрации.** Приведем примеры реконструкции рентгеновских изображений, выполненной с помощью различных методик.

На рис. 1.15 приведены результаты реконструкции РКТ-изображения (модельная томограмма, фантом) [118]. На рис. 1.15а представлено решение (реконструкция фантома) методом (фильтром) Шеппа–Логана (см. (1.13), (1.18)). Этот метод иногда называют методом "интуитивной регуляризации". На рис. 1.15б дано решение методом локальной регуляризации Арсенина [119, с. 55–59] — методом, который может давать в принципе более точные результаты, чем метод (глобальной) регуляризации Тихонова (см. (1.13), (1.19)). Рис. 1.15 является иллюстрацией того, какую различную степень устойчивости томограмм могут давать различные методы.

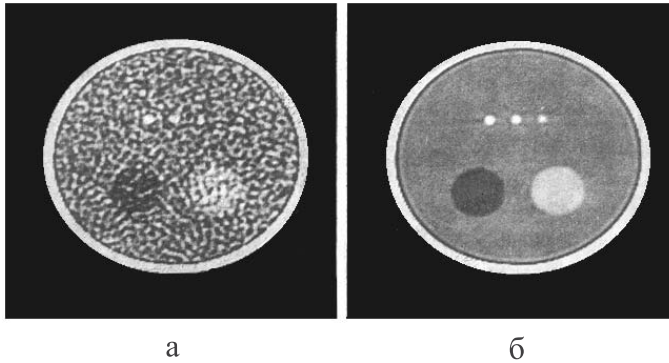


Рис. 1.15. Реконструкция РКТ-изображения (модельный пример, фантом) методом Шеппа–Логана (а) и методом локальной регуляризации Арсенина (б)

На рис. 1.16 приведены РКТ-томограммы [118] одного и того же сечения головного мозга, полученные по методике фирмы ЕМІ (томограф СТ–1010) — см. рис. 1.16а и по методу локальной регуляризации (томограф СРТ–1000) - см. рис. 1.16б.

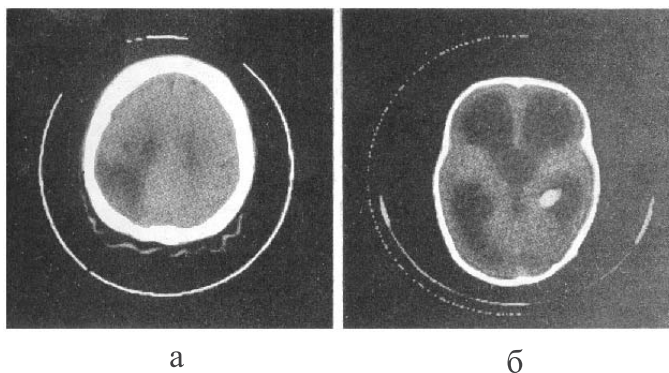


Рис. 1.16. Томограммы сечения мозга по методике ЕМІ (а) и по методу локальной регуляризации (б)

Для этого сечения характерно то, что плотности вещества в соседних участках отличаются лишь на  $\approx 0.5\%$ . Видим, что различные методики могут давать разное разрешение томограмм.

Примеры приведены также в [69, 78], [116, с. 141], [119] и др.

**Задачи с неполными данными.** При выполнении измерений на рентгеновском томографе могут отсутствовать или быть нулевыми значения интенсивностей  $I(I, \theta)$  в некотором диапазоне координат  $l$  или углов  $\theta$ .

Рассмотрим это на примере томографа 3-го поколения (рис. 1.17). На рис. 1.17 и 1.18 представлено два случая, когда данные могут быть неполными.

На рис. 1.17 — случай отсутствия измерений  $I(l, \theta)$  по некоторым техническим причинам. В этом случае для выполнения реконструкции рентгеновского изображения путем решения уравнения (1.16) или (1.20) некоторым методом (например, фильтром Шешпа–Логана, см. (1.13) и (1.18)) потребуется выполнить интерполяцию по углу  $\theta$  или выполнить численные расчеты с переменным шагом дискретизации по  $\theta$ . В результате будет иметь место потеря точности определения плотности  $c(x, y)$ .

На рис. 1.18 представлен случай наличия непрозрачной области в изучаемом объекте, через которую рентгеновские лучи практически не проходят, в результате чего ряд детекторов зафиксирует почти нулевую интенсивность:  $I(l, \theta) \approx 0$ . Тем не менее, в данном случае решение уравнения (1.6) или (1.20) в принципе возможно, однако в некоторой области в сечении объекта будут получены очень большие значения плотности  $c(x, y)$ .

В работе [78, рис. V.5, VI.1, VI.3] даны примеры решений с неполными данными: восстановление по недостаточной выборке направлений  $\theta$ , а также восстановление по недостаточному числу отсчетов по  $l$ ,  $x$  и  $y$  (грубый шаг дискретизации). Примеры показывают, что неполнота данных может приводить к существенным искажениям томограмм.

**Области применения РКТ.** Основная область применения РКТ — медицина [116], а именно, исследование мозга, брюшной полости, грудной клетки, рук и т.д. с целью выявления травм, предраковых опухолей на ранней стадии их развития (размером порядка 1 мм), исследования крови в кровеносных сосудах, деталей анатомического строения сердца, получения динамической картины работы сердца, печени и кровотока с кинематографической регистрацией (в томографах 5-го поколения) и т.д. Другие области применения РКТ: определение трехмерной внутренней структуры технических деталей сложной формы, биологических объектов, плазмы, алмазных выработок, древесины без ее распиловки, контроль узлов реактивных двигателей, просмотр содержимого багажа без его вскрытия на таможне, просвечивание коры Земли (геофизика), оценка распределения плотности электронов вокруг Солнца с использованием естественного томографа Солнце–Земля (астрофизика) и т.д.

### Контрольные задания и вопросы

1. Объяснить суть всех типов томографии (РКТ, МРТ и др.), указав их сходства и различия.
2. Перечислить недостатки обычной рентгенографии.
3. В чем заключается идея РКТ на примере усовершенствованного томографа 1-го поколения?
4. Охарактеризовать пять поколений рентгеновских томографов (задание повышенной трудности).

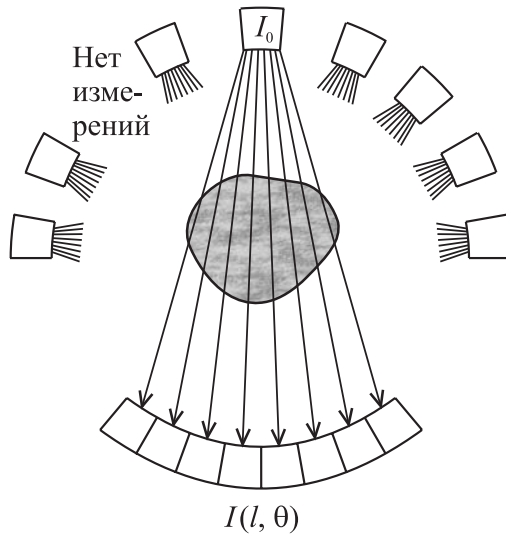


Рис. 1.17. Случай отсутствия измерений в некотором диапазоне углов  $\theta$

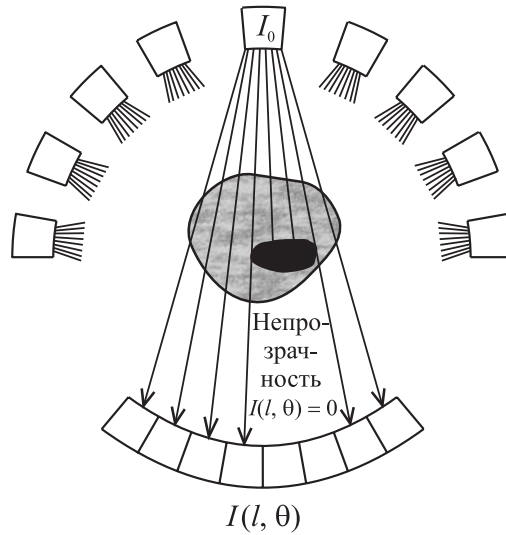


Рис. 1.18. Случай нулевых измерений  $I(l, \theta)$  на некоторых детекторах

5. Записать и истолковать закон Бера. В законе Бера уравнение луча зрения имеет вид:  $x \cos \theta + y \sin \theta = 1$ . Записать вид этого уравнения при  $\theta = 0^\circ, 90^\circ, 180^\circ, 270^\circ$  и указать, как проходит луч зрения.

6. Записать и интерпретировать преобразование и уравнение Радона.

7. Сформулировать и доказать теорему Брейсуэлла о сечении спектра (теорему о центральном, или проекционном слое).

8. Изложить метод преобразования Фурье решения уравнения Радона на основе теоремы Брейсуэлла (задание повышенной трудности).

9. Изложить метод свертки и обратной проекции решения уравнения Радона на основе теоремы Брейсуэлла (задание повышенной трудности).

10. Изложить три варианта корректировки импульсной реакции фильтра  $p(t)$ : фильтры Рамачандрана–Лакшминараяна, Шепша–Логана и на основе метода регуляризации Тихонова (наиболее трудное задание).

11. В чем заключается нестандартность интегрального уравнения Радона? Записать соответствующее ему двухмерное интегральное уравнение Фредгольма I рода типа свертки и привести его решение методом преобразования Фурье (задание повышенной трудности).

12. Дать историческую справку о развитии РКТ.

13. Сформулировать задачи с неполными данными.

## 1.2. Ядерно-магнитно-резонансная томография (МРТ)

Другие названия: ЯМР-томография, МР-томография, магнитно-резонансная томография (МРТ), ЯМР-интроскопия, внутривидение, NMR imaging, NMR tomography [12, 30, 47, 48, 68, 69, 97, 102, 119, 139, 147, 157].

*ЯМР-томография (МРТ)* — это метод отображения внутренней структуры материальных объектов, основанный на явлении ядерного магнитного резонанса (ЯМР) и используемый, главным образом, в медицинских исследованиях. Преимущество МРТ перед РКТ состоит в более высокой разрешающей способности томограмм, их большей контрастности, в возможности получения сечений, необязательно параллельных, и в отсутствии вредного жесткого лучевого воздействия на пациентов и обслуживающий персонал. МРТ по сравнению с РКТ позволяет получить более четкую картину объекта.

**Историческая справка** [68, с. 107–108], [69, с. 41–44], [97, с. 227–240], [102, с. 33]. В 1888 г. Н. Тесла (Хорватия–США) открыл явление вращающегося магнитного поля. В честь Н. Теслы в 1956 г. для измерения магнитной индукции была введена единица Тл (тесла), равная  $10^4$  Гс.

В 1922 г. О. Штерн и В. Герлах экспериментально, а в 1925–26 гг. С. Уленбек и Г. Гаудсмит теоретически доказали, что электрон имеет собственный магнитный момент (наряду с механическим моментом, или спином) с двумя возможными ориентациями — по полю и против поля (рис. 1.19).



Рис. 1.19. Механический момент, или спин электрона (а) и магнитный момент электрона  $\mu$  (б)

В 1937 г. И. Раби (США) впервые наблюдал ЯМР на изолированных ядрах. Он установил, что радиочастотная энергия поглощается или испускается атомными ядрами в магнитном поле  $H_0$  (рис. 1.20), причем эффект поглощения будет особенно силен, когда радиочастота  $\omega$  совпадает с резонансной частотой, или частотой Лармора  $\omega_0 = \gamma H_0$  (см. далее).

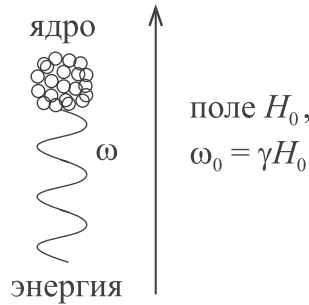


Рис. 1.20. Поглощение энергии атомным ядром в магнитном поле  $H_0$

Сам термин "ядерный магнитный резонанс" (nuclear magnetic resonance) впервые появился в статье К. Гортера (Нидерланды) в 1942 г., хотя К. Гортер отметил, что авторство термина принадлежит И. Раби.

В 1945 г. Е.К. Завойский (СССР) открыл электронный парамагнитный резонанс (ЭПР), а еще раньше (в 1941 г.) проводил опыты (как и К. Гортер) по регистрации ЯМР.

В 1946 г. две группы физиков под руководством Ф. Блоха и Е. Парселла (США), а также Е.К. Завойский наблюдали эффект ЯМР в конденсированных средах.

В 1960 г. В. А. Ивановым (СССР) [47, 48] впервые было предложено использовать ЯМР для определения внутреннего строения материальных объектов.

В 1971 г. Р. Дамадян (США) показал возможность применения ЯМР для исследования опухолей.

В 1972 г. П. Лаутербур (США) [161] предложил использовать градиентные магнитные поля и метод реконструкции изображений по обратным проекциям (заимствованный из РКТ) для получения ЯМР-изображений. Было получено первое двухмерное ЯМР-изображение малых образцов воды, находящихся в двух пробирках. Лаутербур



назвал свой метод "зоигматографией". Позднее этот термин был заменен на "ЯМР-томографию".

В 1976 г. П. Мэнсфилд (Англия) [162] предложил так называемое эхо-планарное отображение (ЕРІ) — эффективную общую ЯМР-методику.

**Поведение магнитных моментов ядер без поля и в поле** [68, с. 108], [69, с. 45], [102, с. 33], [141, с. 7]. Как известно, вещество состоит из молекул (например  $\text{H}_2\text{O}$ ), молекулы — из атомов, а атомы имеют ядра, содержащие протоны ( $p$ ) и нейтроны ( $n$ ). Протон  $p$  и нейрон  $n$  имеют собственные механические вращательные моменты количества движения — спины, а протон имеет еще магнитный момент  $\mu$ . У ядер, имеющих нечетное число протонов  $p$ , есть магнитный момент. Это справедливо, например, для воды, молекула которой  $\text{H}_2\text{O}$  состоит из двух атомов водорода и одного атома кислорода. Ядро атома кислорода имеет четное число (16) протонов  $p$ , следовательно, нулевой магнитный момент  $\mu$ . Однако каждое ядро водорода состоит из одного протона  $p$ , поэтому имеет ненулевой магнитный момент.

В отсутствие внешнего магнитного поля магнитные моменты ядер вещества  $\mu$  (в частности, магнитные моменты двух ядер водорода) ориентированы случайным образом (рис. 1.21а).

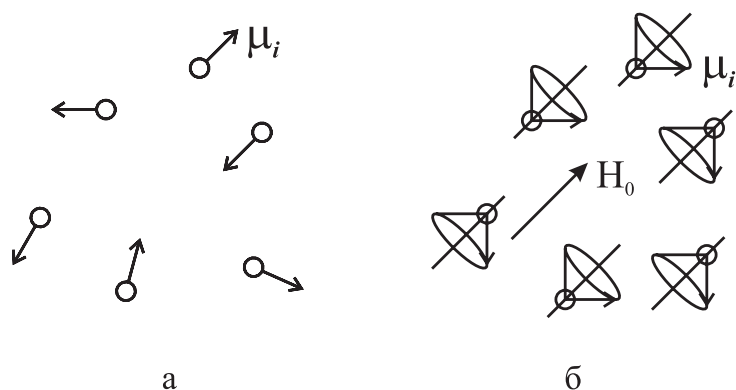


Рис. 1.21. Ориентация магнитных моментов ядер в отсутствие (а) и при наличии (б) магнитного поля  $\mathbf{H}_0$

При включении же магнитного поля  $\mathbf{H}_0$  векторы магнитных моментов ядер  $\mu_i$  будут прецессировать вокруг вектора  $\mathbf{H}_0$ , причем примерно половина по полю и примерно половина против поля (рис. 1.21б). Подробнее о прецессии см. ниже.

*З а м е ч а н и е.* Для характеристики магнитного поля мы будем использовать два обозначения:  $H$  (напряженность) и  $B$  (индукция).

**Уравнение Лармора и ларморова прецессия** [68, с. 109], [102, с. 34], [141, с. 19]. Движение магнитного момента  $\mu$  протона в постоянном магнитном поле  $\mathbf{H}_0$  описывается *дифференциальным уравнением Лармора* (в векторной форме):

$$\frac{d\boldsymbol{\mu}(t)}{dt} = \gamma [\boldsymbol{\mu}(t) \times \mathbf{H}_0], \quad (1.29)$$

где  $\gamma$  — гиромагнитное отношение, равное  $\gamma = e/2m$ , причем  $e$  и  $m$  — заряд и масса протона, а  $[\boldsymbol{\mu}(t) \times \mathbf{H}_0]$  есть векторное произведение. Введем неподвижную (лабораторную) систему координат  $xyz$ , причем ось  $z$  направим вдоль  $\mathbf{H}_0$ . Спроектируем векторное уравнение (1.29) последовательно на  $z$ ,  $x$  и  $y$ , получим (учитывая, что  $H_x = H_y = 0$ ,  $H_z = H_0$ ):

$$\left. \begin{aligned} \frac{d\mu_z(t)}{dt} &= \gamma(\mu_x H_y - \mu_y H_x) = 0, \\ \frac{d\mu_x(t)}{dt} &= \gamma(\mu_y H_z - \mu_z H_y) = \gamma\mu_y H_0, \\ \frac{d\mu_y(t)}{dt} &= \gamma(\mu_z H_x - \mu_x H_z) = -\gamma\mu_x H_0. \end{aligned} \right\} \quad (1.30)$$

Из первого уравнения (1.30) имеем:  $\mu_z(t) = \text{const}$ . А второе и третье уравнения (1.30) образуют систему двух уравнений. Продифференцировав второе уравнение (1.30) по  $t$  и используя третье уравнение, получим:

$$\frac{d^2\mu_x}{dt^2} + (\gamma H_0)^2 \mu_x = 0. \quad (1.31)$$

Уравнение (1.31) есть известное дифференциальное уравнение 2-го порядка (уравнение гармонических колебаний). Его решение

$$\mu_x(t) = A \cos(\gamma H_0 t + \varphi),$$

где  $A$  и  $\varphi$  — некоторые константы (амплитуда и фаза). Аналогично,

$$\mu_y(t) = -A \sin(\gamma H_0 t + \varphi).$$

Введем  $\mu_{\perp}$  — составляющую в плоскости  $x, y$ , равную  $\mu_{\perp} = \sqrt{\mu_x^2 + \mu_y^2}$ . Получим:

$$\mu_{\perp}(t) = A = \text{const}.$$

В результате  $\mu_z(t) = \text{const}$ ,  $\mu_{\perp}(t) = \text{const}$ , значит,  $\theta = \text{const}$ , где  $\theta$  — угол между  $\boldsymbol{\mu}(t)$  и  $\mathbf{H}_0$ , а конец вектора  $\boldsymbol{\mu}_{\perp}(t)$  вращается в плоскости  $xy$  с угловой скоростью

$$\omega_0 = \gamma H_0, \quad (1.32)$$

т. е. в пространстве вектор  $\boldsymbol{\mu}(t)$  прецессирует по конусу прецессии вокруг  $\mathbf{H}_0$  с угловой скоростью (1.32) (см. рис. 1.22). Данная прецессия является проявлением гироскопического эффекта и напоминает движение механического волчка (гироскопа).

Прецессия магнитных моментов протонов в магнитном поле — это основной процесс, характерный для ЯМР.

*Определение.* Угловая скорость  $\omega_0 = \gamma H_0$  называется *угловой скоростью (частотой) ларморовой прецессии*.

Если же ввести систему координат, вращающуюся вокруг оси  $z$  с угловой скоростью  $\omega_0 = \gamma H_0$ , то получим в ней:  $\mu_z(t) = \text{const}$ ,  $\theta(t) = \text{const}$ ,  $\mu_x(t) = A \cos \varphi = \text{const}$ ,  $\mu_y(t) = -A \sin \varphi = \text{const}$ ,  $\mu_{\perp}(t) = \text{const}$ .

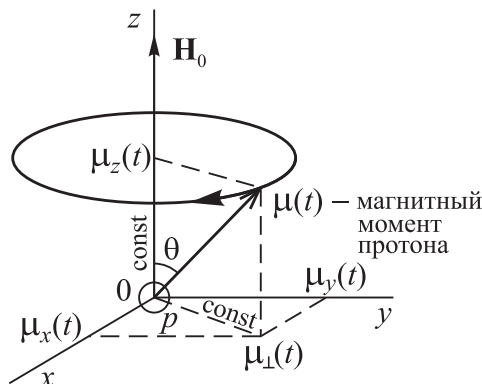


Рис. 1.22. Прецессия магнитного момента  $\mu(t)$  протона вокруг магнитного поля  $\mathbf{H}_0$

Однако изложенный подход является классическим. С позиций же квантовой механики магнитный момент  $\mu$  не прецессирует по конусу с определенной частотой, а следует говорить о вероятности его нахождения на поверхности конуса.

**Ансамбль протонов** [102, с. 36]. Рассмотрим некоторое большое количество протонов — ансамбль (популяцию) протонов. Практически это миллиарды или триллионы протонов в пределах, например,  $1 \text{ мм}^3$  вещества. Согласно квантовой механике, все протоны, входящие в ансамбль, имеют лишь  $m$  конусов прецессии, причем  $m = 2I + 1$ , где  $I = 0, 1, \dots, 6$  — так называемый ядерный спин, направленный как вдоль поля  $\mathbf{H}_0$ , так и против него.

Введем в рассмотрение суммарный вектор ансамбля протонов

$$\mathbf{M} = \sum_i \mu_i, \quad (1.33)$$

называемый вектором ядерной намагниченности [141, с. 32]. Казалось бы, в силу симметрии, должно быть одинаковое количество магнитных моментов  $\mu_i$ , направленных по полю и против него, и вектор  $\mathbf{M}$  должен быть равен нулю. Однако несимметричная квантовая механика показывает, что отношение числа протонов, прецессирующих по полю, к числу протонов, прецессирующих против него, чуть больше единицы и равно [68, с. 118], [102, с. 36]

$$N_{\text{отн}} = e^{\Delta E/kT}, \quad (1.34)$$

где  $\Delta E = \gamma \hbar H_0$ ,  $\hbar = h/2\pi$ ,  $h$  и  $k$  — постоянные Планка и Больцмана,  $T$  — абсолютная температура. Если, например,  $T = 25^\circ\text{C} = 298 \text{ К}$ ,  $H_0 = 1 \text{ Тл}$ , то  $N_{\text{отн}} \approx 1.000003 = 1 + 3 \cdot 10^{-6}$ , т. е. из миллиона протонов лишь на 3 протона будет больше прецессировать по полю, чем против поля. В результате вектор ядерной намагниченности ансамбля протонов  $\mathbf{M}$  будет направлен по полю  $\mathbf{H}_0$ :

$$\mathbf{M} = \chi \mathbf{H}_0, \quad (1.35)$$

где  $\chi$  называется ядерной магнитной восприимчивостью [141, с. 34]. Однако намагниченность  $M$  очень мала, поэтому долго не могли обнаружить эффект ядерной на-

магнитности. Более того, вся ЯМР-томография, по существу, основана на отличии  $\chi$  от нуля, другими словами, на несимметричной ориентации магнитных моментов протонов по полю  $\mathbf{H}_0$  и против него, хотя и очень мал тот эффект. Поэтому нижеизложенные методы ЯМР-томографии связаны с очень слабыми сигналами от вещества (ЯМР-сигналами) и требуется охлаждение томографа жидким гелием, высокая степень однородности поля  $H_0$  ( $\Delta H/H_0 \sim 10^{-6}$ ) и т. д.

**Движение магнитного момента протона в постоянном и переменном магнитных полях** [102, с. 36–38], [141, с. 23–26]. Пусть помимо постоянного магнитного поля  $\mathbf{H}_0$  (вдоль  $z$ ) действует перпендикулярно ему переменное поле, *осциллирующее* вдоль  $x$ :  $H_x = H_x(t) = 2H_1 \cos \omega t$ , причем  $H_1 \ll H_0$  (на практике  $\omega \sim 1\text{--}10$  Мгц — радиочастотный диапазон). Можно показать, что  $H_x(t)$  можно представить в виде суммы двух *вращающихся* в разных направлениях полей  $\mathbf{H}^{(1)}(t)$  и  $\mathbf{H}^{(2)}(t)$  с одинаковой частотой (угловой скоростью)  $\omega$  и одинаковой амплитудой  $H_1$  (см. рис. 1.23).

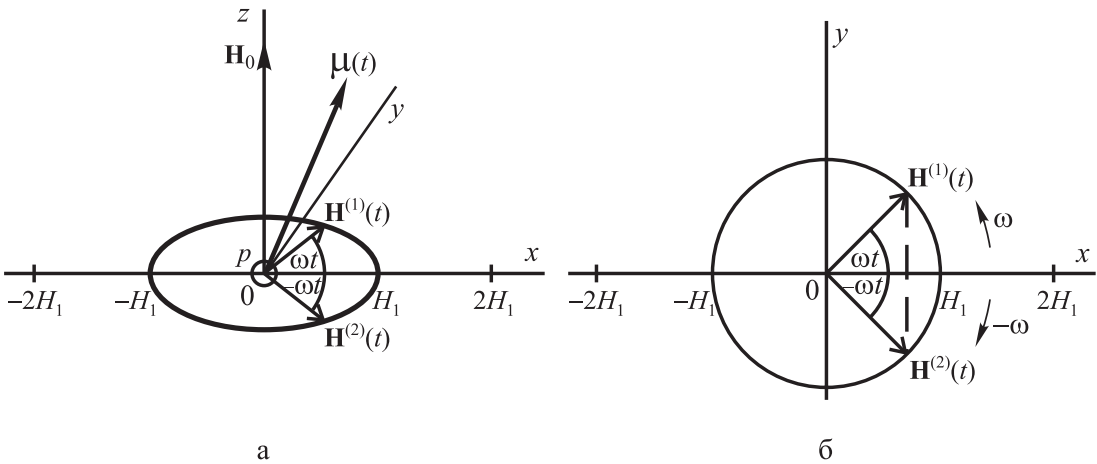


Рис. 1.23. Магнитный момент протона  $\mu$  в постоянном и переменном полях, а — пространственная картина, б — картина в плоскости  $xy$

Действительно, сумма проекций  $\mathbf{H}^{(1)}(t)$  и  $\mathbf{H}^{(2)}(t)$  на ось  $x$  равна

$$H_x^{(1)}(t) + H_x^{(2)}(t) = H_1 \cos \omega t + H_1 \cos \omega t = 2H_1 \cos \omega t = H_x(t), \quad (1.36)$$

а сумма проекций  $\mathbf{H}^{(1)}(t)$  и  $\mathbf{H}^{(2)}(t)$  на  $y$  равна

$$H_y^{(1)}(t) + H_y^{(2)}(t) = H_1 \sin \omega t - H_1 \sin \omega t = 0. \quad (1.37)$$

Постоянному магнитному полю напряженности  $H_0$  соответствует частота  $\omega_0 = \gamma H_0$ , а одному из вращающихся полей  $H^{(1)}(t)$  или  $H^{(2)}(t)$  — частота  $\omega_1 = \gamma H_1$ .

Определение. Частота  $\omega_0 = \gamma H_0$  называется *частотой прецессии* (или угловой скоростью ларморовой прецессии), частота  $\omega_1 = \gamma H_1 \ll \omega_0$  — *частотой нутации*, а  $\omega$  — *частотой вращения полей*  $H^{(1)}(t)$  и  $H^{(2)}(t)$ .

Определение. Разность частот  $\Delta\omega = \omega_0 - \omega$  называется *расстройкой по частоте*.

Если  $\Delta\omega = 0$ , т. е. одно из полей  $H^{(1)}(t)$  или  $H^{(2)}(t)$  вращается в такт с прецессией протона вокруг  $H_0$ , то имеет место *резонанс*:  $\omega_0 = \omega$ . В результате магнитный момент  $\mu$  будет совершать одновременно *две прецессии*: вокруг  $H_0$  (или оси  $z$ ) с угловой скоростью *прецессии*  $\omega_0 = \gamma H_0$  и вокруг  $H_1$  (или оси  $x$ ) с частотой *нутаии*  $\omega_1 = \gamma H_1 \ll \omega_0$ , а именно, в лабораторной (неподвижной) системе координат конец вектора  $\mu$  будет выписывать густую *спираль на поверхности шара* радиуса  $|\mu|$  (рис. 1.24). При этом (при  $\Delta\omega = 0$ ) угол  $\theta$  (угол между постоянным полем  $\mathbf{H}_0$  и магнитным моментом  $\mu$  – угол нутации Эйлера) будет изменяться за время действия переменного поля  $H_x(t)$  от некоторого минимального значения  $\theta_1$  до некоторого максимального значения  $\theta_2$ .

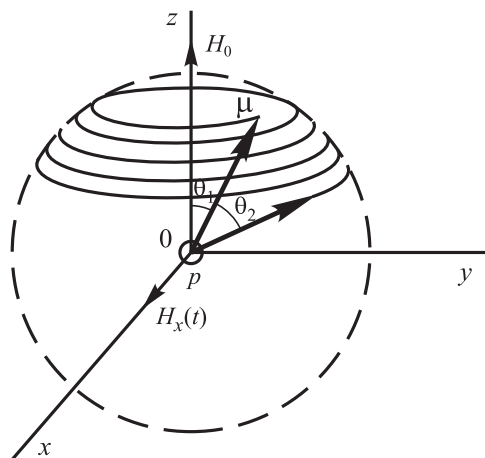


Рис. 1.24. Спираль на поверхности шара, выписываемая концом вектора  $\mu$  в постоянном  $H_0$  и переменном  $H_x(t)$  полях

Если  $\Delta\omega = 0$ , т. е.  $\omega = \omega_0$ , то будет иметь место *ядерный магнитный резонанс* (ЯМР), или избирательное поглощение электромагнитной энергии веществом. Как будет показано далее, это приведет к тому, что само вещество будет излучать электромагнитную энергию в виде так называемого *эхо-сигнала* (или *ЯМР-сигнала*), математическая обработка которого (методом Кумара–Велти–Эрнста или иным методом, см. дальше) позволит найти распределение плотности вещества в объекте  $s(x, y, z)$ .

**Уравнения Блоха** [68, с. 119], [69, с. 49], [79, с. 12], [102, с. 38], [141, с. 42],. Опять рассмотрим *ансамбль протонов* и проанализируем случай, когда постоянное магнитное поле  $\mathbf{V}_0$  (старое значение) мгновенно отключается и сразу же включается новое постоянное поле  $\mathbf{V}_z$  (ось  $z$  направим вдоль нового поля). Обозначим угол между  $\mathbf{V}_0$  и  $\mathbf{V}_z$  через  $\theta$ .

Согласно (1.35), в начальный момент времени (непосредственно перед переключением полей) вектор ядерной намагниченности ансамбля протонов  $\mathbf{M}$  равен  $\mathbf{M}(0) = \chi\mathbf{V}_0$ , а при  $t \rightarrow \infty$   $\mathbf{M}(t) = \mathbf{M}_z = \chi\mathbf{V}_z$ . Спрашивается, насколько быстро и по какой траектории вектор  $\mathbf{M}$  перейдет из начального состояния  $\chi\mathbf{V}_0$  в новое, конечное состояние  $\chi\mathbf{V}_z$ ? Ответ на этот вопрос дает *феноменологическое* (описательное, эмпирическое) *дифференциальное уравнение Блоха*, которое в векторной форме в лабораторной

(неподвижной) системе координат записывается в виде [68, с. 119], [141, с. 42]:

$$\frac{d\mathbf{M}(t)}{dt} = \gamma [\mathbf{M}(t) \times \mathbf{B}_z] - \frac{M_z(t) - M_z}{T_1} \mathbf{e}_z - \frac{M_x(t) \mathbf{e}_x + M_y(t) \mathbf{e}_y}{T_2^*}, \quad (1.38)$$

где  $\mathbf{e}_x, \mathbf{e}_y, \mathbf{e}_z$  — единичные векторы вдоль  $x, y, z$ .

Первое слагаемое в правой части (1.38)  $\gamma[\mathbf{M}(t) \times \mathbf{B}_z]$  по аналогии с правой частью уравнения (1.29) означает *прецессию* вектора  $\mathbf{M}(t)$  вокруг поля  $\mathbf{B}_z$  с угловой скоростью  $\omega = \gamma B_z$  (по аналогии с (1.32)). Следующие слагаемые в правой части (1.38) описывают так называемые *процессы релаксации* (ослабления, уменьшения, затухания, перехода в новое положение), а именно, спин-решетчатую и спин-спиновую релаксации [97, с. 34, 42].

*Спин-решетчатая*, или *продольная релаксация*, характеризуемая временем  $T_1$ , обусловлена энергетическими обментами между спинами (протонами) и средой (решеткой), в которой они находятся. *Спин-спиновая*, или *поперечная релаксация*, характеризуемая временем  $T_2$ , обусловлена взаимодействиями между спинами. Эти взаимодействия создают *локальные поля*, т.е. неоднородности поля  $\mathbf{B}$ , что ведет к неодинаковости угловых ларморовых скоростей  $\omega$ , а значит, к *расфазированию* спинов и потере резонанса в плоскости  $xy$  [30, с. 9] (рис. 1.25).

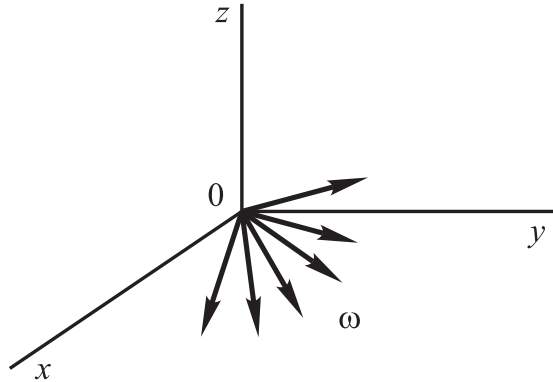


Рис. 1.25. Неодинаковость угловых скоростей  $\omega$  и расфазирование спинов в плоскости  $xy$

Кроме того, имеет место *техническая неоднородность магнитного поля*  $\Delta B$ , усиливающая расфазирование. В этом случае вместо  $T_2$  нужно использовать новое время поперечной релаксации  $T_2^*$  [68, с. 131]:

$$\frac{1}{T_2^*} = \frac{1}{T_2} + \gamma \frac{\Delta B}{2},$$

в результате  $T_2^* \leq T_2$ , т.е. релаксация (расфазирование, потеря резонанса) в плоскости  $xy$  происходит быстрее при  $\Delta B \neq 0$ , чем при  $\Delta B = 0$ .

Остановимся на *решении и анализе уравнения Блоха*.

Во вращающейся (с угловой скоростью  $\omega = \gamma B_z$ ) системе координат прецессия вектора  $\mathbf{M}(t)$  вокруг  $\mathbf{B}_z$  исключается, т. е. первое слагаемое в правой части будет отсутствовать и уравнение Блоха в векторной форме запишется в виде:

$$\frac{d\mathbf{M}(t)}{dt} = -\frac{M_z(t) - M_z}{T_1} \mathbf{e}_z - \frac{M_x(t) \mathbf{e}_x + M_y(t) \mathbf{e}_y}{T_2^*}. \quad (1.39)$$

Проектируя векторное уравнение (1.39) на оси  $z$ ,  $x$  и  $y$ , получим три самостоятельных *дифференциальных уравнения Блоха*:

$$\begin{aligned} \frac{dM_z(t)}{dt} &= -\frac{M_z(t) - M_z}{T_1}, \\ \frac{dM_x(t)}{dt} &= -\frac{M_x(t)}{T_2^*}, \\ \frac{dM_y(t)}{dt} &= -\frac{M_y(t)}{T_2^*}. \end{aligned} \quad (1.40)$$

Геометрически переход вектора  $\mathbf{M}$  из начального положения  $\mathbf{M}_0$  в конечное положение  $\mathbf{M}_z$  условно отображен на рис. 1.26, где  $\beta_0$  — начальная фаза.

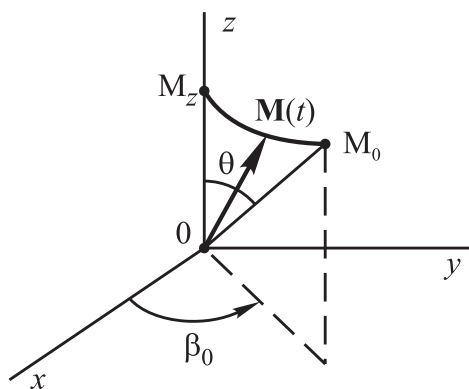


Рис. 1.26. Переход вектора  $\mathbf{M}$  из начального положения  $\mathbf{M}_0$  в конечное положение  $\mathbf{M}_z$  во вращающейся системе координат

Проинтегрируем каждое из уравнений (1.40) от начального момента  $t = 0$  до те-

кущего  $t$ :

$$\begin{aligned} \int_{M_0 \cos \theta}^{M_z(t)} \frac{dM_z(t')}{M_z(t') - M_z} &= -\frac{1}{T_1} \int_0^t dt', \\ \int_{M_0 \sin \theta \cos \beta_0}^{M_x(t)} \frac{dM_x(t')}{M_x(t')} &= -\frac{1}{T_2^*} \int_0^t dt', \\ \int_{M_0 \sin \theta \sin \beta_0}^{M_y(t)} \frac{dM_y(t')}{M_y(t')} &= -\frac{1}{T_2^*} \int_0^t dt', \end{aligned}$$

откуда

$$\ln \frac{M_z(t) - M_z}{M_0 \cos \theta - M_z} = -\frac{t}{T_1}, \quad \ln \frac{M_x(t)}{M_0 \sin \theta \cos \beta_0} = -\frac{t}{T_2^*}, \quad \ln \frac{M_y(t)}{M_0 \sin \theta \sin \beta_0} = -\frac{t}{T_2^*},$$

или, добавив  $M_{\perp} \equiv M_{xy}(t) = \sqrt{M_x^2(t) + M_y^2(t)}$  — проекцию  $\mathbf{M}(t)$  на плоскость  $xy$ ,

$$\begin{aligned} M_z(t) &= M_z + (M_0 \cos \theta - M_z) e^{-t/T_1}, \\ M_x(t) &= M_0 \sin \theta \cos \beta_0 e^{-t/T_2^*}, \\ M_y(t) &= M_0 \sin \theta \sin \beta_0 e^{-t/T_2^*}, \\ M_{\perp}(t) &= M_0 \sin \theta e^{-t/T_2^*}. \end{aligned} \tag{1.41}$$

Выражения (1.41) описывают поведение проекций вектора  $\mathbf{M}(t)$  во вращающейся системе координат. Видим, что  $M_z(0) = M_0 \cos \theta$ ,  $M_z(\infty) = M_z$ ,  $M_{\perp}(0) = M_0 \sin \theta$ ,  $M_{\perp}(\infty) = 0$ , т.е. проекция  $M_z(t)$  по экспоненте переходит от начального значения  $M_0 \cos \theta$  к конечному значению  $M_z$ , а проекция  $M_{\perp}(t)$  (поперечная намагниченность) также по экспоненте переходит от  $M_0 \sin \theta$  к нулевому значению. При этом переход осуществляется в принципе за бесконечное время. Однако отметим, что  $e^{-1} \approx 0.37$ ,  $e^{-2} \approx 0.135$ ,  $e^{-3} \approx 0.050$ ,  $e^{-4} \approx 0.018$ ,  $e^{-5} \approx 0.0067$ ,  $e^{-6} \approx 0.0025$ ,  $e^{-7} \approx 0.00091 \approx 10^{-3}$ , ... Это говорит о том, что при  $t \approx 7T_1$  или  $7T_2^*$  экспоненты в (1.41) выходят на уровень  $\approx 10^{-3}$ , т.е. практически обнуляются. Другими словами, процессы продольной релаксации (см. первое выражение (1.41)) проходят практически за время  $t \approx 7T_1$ , а процессы поперечной релаксации (см. последнее выражение (1.41)) — за время  $t \approx 7T_2^*$ , после чего  $M_{\perp}$  практически обнуляется и вектор  $\mathbf{M}$  переходит в статическое положение, приняв значение  $M_z = \chi B_z$ .

В лабораторной (неподвижной) системе координат нужно учесть прецессию  $\mathbf{M}$  вокруг  $\mathbf{B}_z$  и выражения (1.41) примут вид:

$$\begin{aligned} M_z(t) &= M_z + (M_0 \cos \theta - M_z) e^{-t/T_1}, \\ M_x(t) &= M_0 \sin \theta \cos \beta(t) e^{-t/T_2^*}, \\ M_y(t) &= M_0 \sin \theta \sin \beta(t) e^{-t/T_2^*}, \\ M_{\perp}(t) &= M_0 \sin \theta e^{-t/T_2^*}, \end{aligned} \tag{1.42}$$



где  $\beta(t) = \beta_0 + \omega t = \beta_0 + \gamma B_z t$  — изменение фазы.

Из выражений (1.42) следует, что конец вектора  $\mathbf{M}$  будет двигаться по спирали от  $\mathbf{M}_0$  (при  $t = 0$ ) до  $\mathbf{M}_z$  (при  $t \rightarrow \infty$ , а практически при  $t \approx 7T_1$ ), причем проекция  $M_{\perp}(t)$  будет совершать вращение в плоскости  $xy$  с угловой скоростью  $\omega = \gamma B_z$ , убывая от  $M_{\perp}(0) = M_0 \sin \theta$  до  $M_{\perp} = 0$  (при  $t \rightarrow \infty$ , а практически при  $t \approx 7T_2^*$ ).

На рис. 1.27 отображен спиралевидный переход вектора ядерной намагниченности ансамбля протонов  $\mathbf{M}$  из начального положения  $\mathbf{M}_0$  в конечное, стационарное положение  $\mathbf{M}_z$  для двух случаев:  $M_z(0) < M_z$  (рис. 1.27а) и  $M_z(0) > M_z$  (рис. 1.27б).

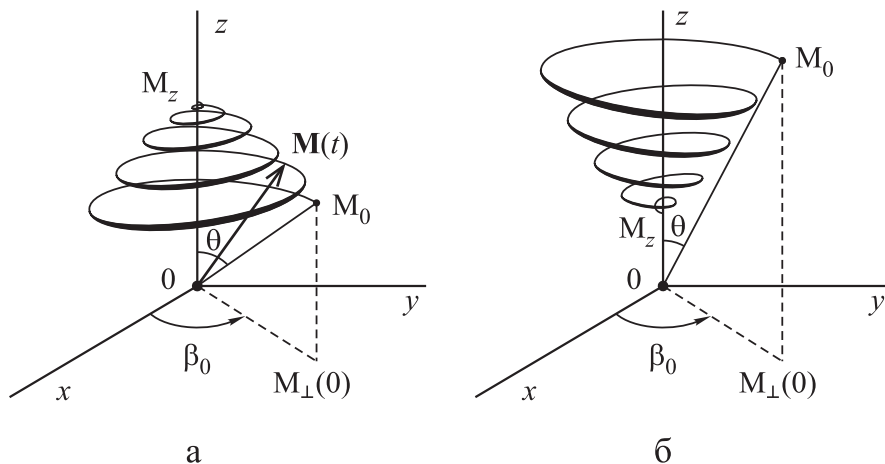


Рис. 1.27. Два примера перехода вектора  $\mathbf{M}$  из начального положения  $\mathbf{M}_0$  в конечное положение  $\mathbf{M}_z$  в неподвижной системе координат

**Некоторые численные данные.** В табл. 1.1, 1.2 и 1.3 приведены значения гиромагнитного отношения  $\gamma$ , времени продольной релаксации  $T_1$  и процентное содержание воды в различных тканях организма [47, 48].

Заметим, что в системе СИ единица индукции магнитного поля  $B$  есть 1 Тл (1 тесла, равная  $10^4$  Гс, где Гс — единица индукции в системе СГС), а единица напряженности  $H$  есть 1 А/м.

Напомним, что частота (угловая скорость) ларморовой прецессии магнитного момента протона или ядра равна  $\omega = \gamma B$ . Из табл. 1.1 следует, что, например, при  $B = 1$  Тл частота ларморовой прецессии ядра водорода  $\text{H}^2$  равна  $\omega = 267.5$  млн рад/сек или 42.58 млн оборот/сек.

Таблица 1.1

ядро	$\gamma$ , МГц/Тл	$\gamma/2\pi$
водорода $\text{H}^2$	267.5	42.58
углерода $\text{C}^{13}$	67.1	10.7
фосфора $\text{P}^{31}$	108.1	17.2

Далее, табл. 1.1 показывает, что значение  $\gamma$ , а значит, ларморовы частоты различных веществ могут быть весьма различны. Поэтому, исследуя, например, распределение водорода (а, значит, воды) в организме в районе частоты  $\omega = \gamma B$ , где  $\gamma = 267.5$  МГц/Тл, мы не будем получать мешающиеся данные от других веществ (углерода, фосфора и т. д.).

Таблица 1.2

орган	$T_1$ , сек (здоровая ткань)	$T_1$ , сек (рак)
легкое	0.73	1.1
кишечник	0.68	1.15
желудок	0.75	1.2
печень	0.59	0.8
селезенка	0.7	1.1

Табл. 1.2 показывает, что время продольной релаксации  $T_1$  весьма различно у здоровой и раковой ткани (это впервые показал Р. Дамадиан в 1971 г.). Поэтому, определяя  $T_1$  некоторого органа или его части, можно поставить диагноз: здоровая ткань, предраковая ткань, рак. Время  $T_1$  можно определить по огибающей мнимой части ЯМР-сигнала или эхо-сигнала  $\text{Im } s(t)$ , пропорциональной  $e^{-t/T_1}$  (см. дальше), или другими способами [68, с. 129–130].

Таблица 1.3

ткань	% воды
серое вещество мозга	83
почки	81
сердце	80
мышцы, легкие	78
печень	71
кожа	68
кость	12

Табл. 1.3 показывает, что в тканях организма достаточно много воды, а значит, протонов, что создает благоприятные условия для реконструкции ЯМР-изображений на основе определения плотности протонов  $s_z(x, y)$  (см. дальше).

Кратко остановимся на значениях времен продольной  $T_1$  и поперечной  $T_2$  релаксации для различных веществ. В жидкостях  $T_1$  и  $T_2$  обычно близки друг другу. Однако

их кристаллизация (например, превращение жидкости в лед) приводит к значительному уменьшению  $T_2$ . В чистых диамагнитных кристаллах (намагничивающихся в магнитном поле в противоположном ему направлении)  $T_1 \sim 1 - 10$  час. А у парамагнитных веществ (намагничивающихся по направлению поля)  $T_1$  значительно меньше, например, у парамагнитных жидких растворов  $T_1 \sim 10^{-3} - 10^{-4}$  сек. В металлах при сверхнизких температурах ( $1 - 10$ К)  $T_1 \sim 10^{-3} - 10$  сек. Видим, что диапазон значений  $T_1$  и  $T_2$  весьма велик — от долей мсек до нескольких часов.

**ЯМР-сигнал** [68, с. 121], [102, с. 41]. Важным обстоятельством является то, что пока поперечная намагниченность ансамбля протонов  $M_{\perp} \equiv M_{xy}$  (см. (1.42) и рис. 1.27) отлична от нуля, ансамбль протонов излучает электромагнитный радиочастотный сигнал  $s(t)$ . Однако  $M_{\perp}(t)$  убывает с ростом  $t$ , как это видно из (1.42), и это убывание обусловлено процессами релаксации, приводящими к расфазированию протонов. В пределе (при  $t \rightarrow \infty$ , а практически при  $t \approx 7T_2^*$ ) наступает равновесное состояние, когда намагниченность  $\mathbf{M}$  будет направлена вдоль  $\mathbf{M}_z$  и  $M_{\perp} = 0$ , т.е. не будет прецессии вектора  $\mathbf{M}$  вокруг  $\mathbf{M}_z$  и сигнала от ансамбля протонов. Чтобы получить сигнал, нужно отклонить вектор  $\mathbf{M}$  от  $\mathbf{M}_z$  — в этом случае будет  $M_{\perp} \neq 0$ . Это — основа ЯМР-измерений. А зная сигнал  $s(t)$ , можно определить плотность протонов  $c_z(x, y)$  (см. далее).

Отклонить вектор  $\mathbf{M}$  от  $\mathbf{M}_z$  можно с помощью так называемого *импульсного метода*. Согласно нему, создается короткий мощный (радиочастотный) импульс (ВЧ-импульс возбуждения [68, с. 124]) в направлении, непараллельном  $\mathbf{B}_z$ , лучше всего по углом  $90^\circ$  к  $\mathbf{B}_z$  ( $\pi/2$ -импульс резонансной частоты  $\omega = \gamma B_z$ ), например, вдоль  $y$  (без отключения  $B_z$ ). В этом случае конец вектора  $\mathbf{M}$  будет совершать спиралевидное отклонение от оси  $z$  (выполняя две прецессии — вокруг осей  $z$  и  $y$ , ср. рис. 1.24) с появлением компоненты  $M_{\perp}$  (см. рис. 1.28).

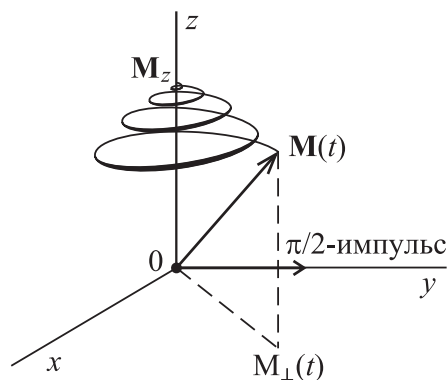


Рис. 1.28. Отклонение вектора  $\mathbf{M}$  от  $\mathbf{M}_z$  под действием  $\pi/2$ -импульса

От ансамбля протонов будет исходить затухающий ЯМР-сигнал  $s(t)$ , или сигнал *спада свободной индукции* — ССИ [68, с. 121] (Free Induction Decay — FID [97, 157]). Проанализируем его подробнее.

Амплитуда ВЧ-импульса приблизительно в  $10^6 - 10^9$  раз больше, чем амплитуда ЯМР-сигнала. Продолжительность импульса почти всегда много короче, чем продол-

жительность сигнала ( $\approx 7T_2^*$ ).

ЯМР-сигнал  $s(t)$  имеет составляющие по  $x$  и по  $y$ . Поэтому  $s(t)$  можно записать в комплексной форме:  $s(t) = s_x(t) + i s_y(t)$ . После действия (короткого)  $\pi/2$ -импульса обе составляющие  $s_x(t)$  и  $s_y(t)$  будут убывать так же, как убывают  $M_x$  и  $M_y$  в (1.42). Запишем закон их убывания в упрощенной форме (при  $t \geq 0$ ):

$$\begin{aligned} \operatorname{Re} s(t) &\equiv s_x(t) = \cos(\omega_z t) e^{-t/T}, \\ \operatorname{Im} s(t) &\equiv s_y(t) = \sin(\omega_z t) e^{-t/T}, \end{aligned} \quad (1.43)$$

где  $\omega_z = \gamma B_z$ ,  $T = T_2^*$ ,  $\beta_0 = 0$  (обычно закон убывания сигнала ССИ более сложен, чем (1.43) [97, с. 17]).

Большой интерес представляет (в первую очередь, для ЯМР-спектроскопии [79]) спектр  $S(\omega)$  ЯМР-сигнала  $s(t)$ . Чтобы получить его, выполним преобразование Фурье (ПФ):

$$\begin{aligned} S(\omega) &= \int_{-\infty}^{\infty} s(t) e^{-i\omega t} dt = \\ &= \int_0^{\infty} [\cos(\omega_z t) e^{-t/T} + i \sin(\omega_z t) e^{-t/T}] \cdot [\cos(\omega t) - i \sin(\omega t)] dt = \\ &= \int_0^{\infty} [\cos(\omega_z t) \cos(\omega t) + \sin(\omega_z t) \sin(\omega t)] e^{-t/T} dt + \\ &+ i \int_0^{\infty} [-\cos(\omega_z t) \sin(\omega t) + \sin(\omega_z t) \cos(\omega t)] e^{-t/T} dt. \end{aligned} \quad (1.44)$$

Интегрирование в (1.44) дает:

$$S(\omega) = \operatorname{Re} S(\omega) + i \operatorname{Im} S(\omega),$$

где (ср. [68, с. 122])

$$\operatorname{Re} S(\omega) = \frac{T}{1 + (\omega - \omega_z)^2 T^2}, \quad (1.45)$$

$$\operatorname{Im} S(\omega) = -\frac{(\omega - \omega_z) T^2}{1 + (\omega - \omega_z)^2 T^2}. \quad (1.46)$$

Вещественную часть спектра  $\operatorname{Re} S(\omega)$  называют *сигналом поглощения*, а мнимую часть  $\operatorname{Im} S(\omega)$  — *сигналом дисперсии*, или *рассеяния* [68, с. 121]. На рис. 1.29 приведены  $\operatorname{Re} s(t)$ ,  $\operatorname{Im} s(t)$ ,  $\operatorname{Re} S(\omega)$  и  $\operatorname{Im} S(\omega)$  при  $T = 1$ ,  $\omega_z = 25\pi/3$  (ср. [68, с. 121], [141, с. 46]).

Часто [79] вычисляют модуль спектра  $|S(\omega)| = \sqrt{\operatorname{Re}^2 S(\omega) + \operatorname{Im}^2 S(\omega)}$ , чтобы вместо двух графиков  $\operatorname{Re} S(\omega)$  и  $\operatorname{Im} S(\omega)$  получить один график  $|S(\omega)|$ . Однако при этом теряется информация о фазе. На рис. 1.30 приведен  $|S(\omega)|$  по данным рис. 1.29.

**Основные импульсные последовательности** [12], [68, с. 122], [69, с. 58], [97, с. 33], [102, с. 43], [157]. Рассмотренный выше ЯМР-сигнал, или сигнал спада свободной индукции является слабым и затухающим из-за расфазирования протонов (см. рис. 1.25), обусловленного процессами релаксации. Такой сигнал практически

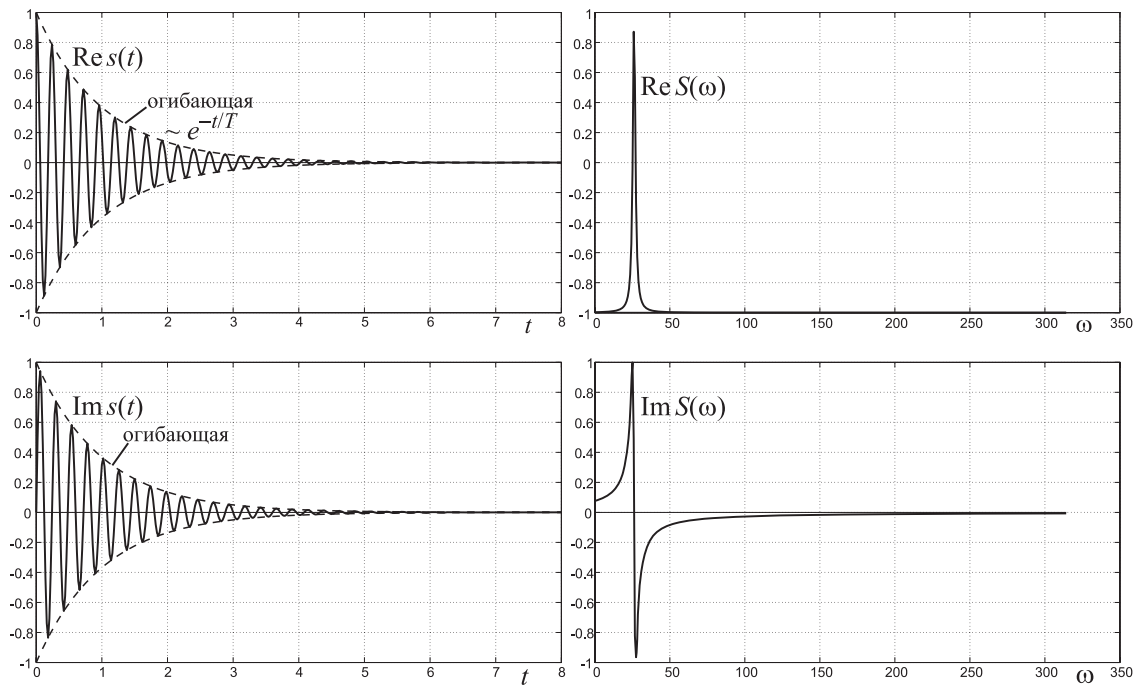


Рис. 1.29. Вещественная  $\text{Re } s(t)$  и мнимая  $\text{Im } s(t)$  части ЯМР-сигнала  $s(t)$ , а также вещественная  $\text{Re } S(\omega)$  и мнимая  $\text{Im } S(\omega)$  части его спектра  $S(\omega)$

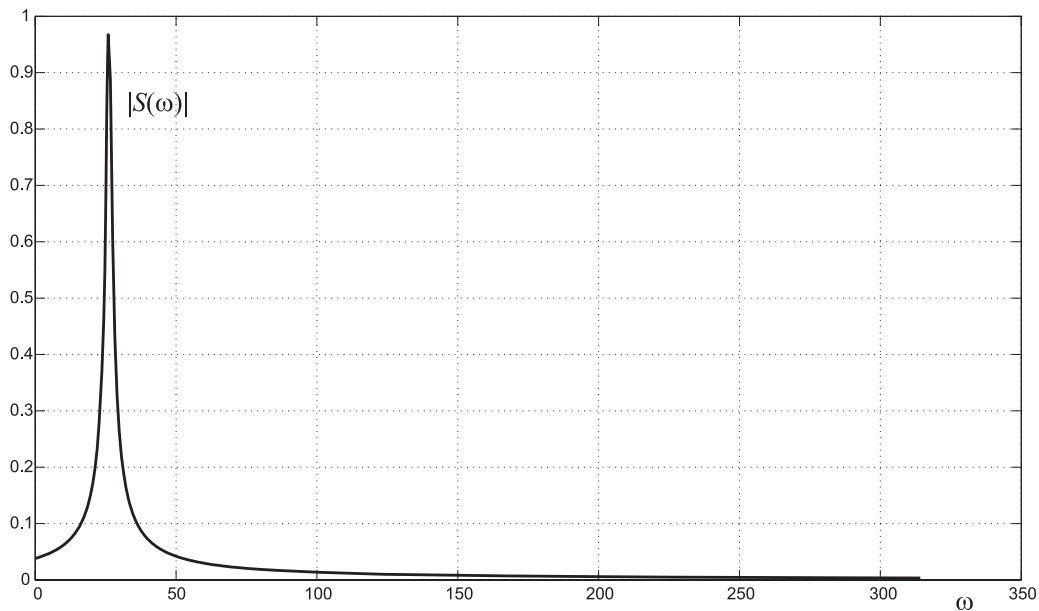


Рис. 1.30. Модуль спектра  $|S(\omega)|$

нельзя зарегистрировать на фоне аппаратурных шумов. Чтобы получить достаточно сильный ЯМР-сигнал, используют различные *импульсные последовательности* (ИП). Рассмотрим некоторые из них.

На рис. 1.31 показана так называемая *импульсная последовательность для регистрации спин-эха* [68, с. 125–126], или *спин-эхо-ИП* (SE, spin echo) [69, с. 58–59], [97, с. 43], [157, Fig. 12]. Заметим, что часто импульсы отображают в виде прямоугольников (как на рис. 1.31) или в виде огибающих [69, с. 59], хотя импульс имеет тонкую структуру (см. далее рис. 1.36).

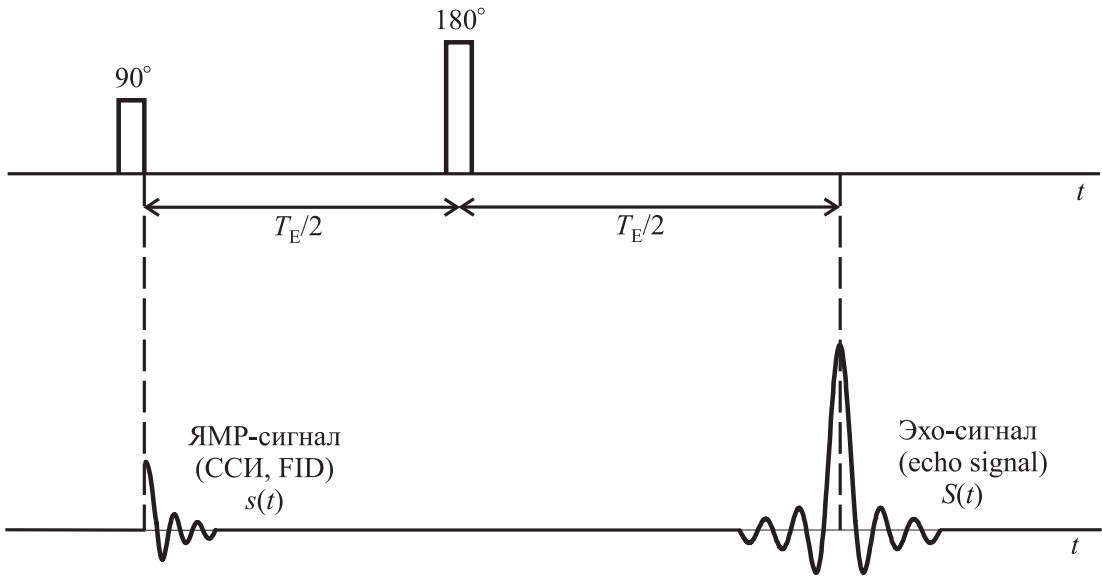


Рис. 1.31. Спин-эхо-ИП.

$90^\circ$ -ный импульс создает компоненту  $M_{xy}$  и ЯМР-сигнал  $s(t)$ , однако он слаб из-за расфазирования, а  $180^\circ$ -ный импульс создает рефазирование и сильный эхо-сигнал  $S(t)$

В данном варианте создается  $90^\circ$ -ный импульс (или  $\pi/2$ -импульс) – короткий мощный ВЧ-импульс под углом  $90^\circ$  к  $\mathbf{B}_z$ . В результате от ансамбля протонов пойдет ЯМР-сигнал (ССИ, FID)  $s(t)$ . Однако это будет слабый сигнал из-за того, что процессы релаксации создают неоднородные, локальные поля и угловые скорости ларморовой прецессии протонов становятся неодинаковыми, т.е. возникает эффект расфазирования. Чтобы получить достаточно сильный, надежно измеряемый сигнал, по истечении некоторого времени  $T_E/2$  создается еще один импульс ( $180^\circ$ -ный импульс) – импульс, сдвинутый на  $180^\circ$  (противоположный  $\mathbf{B}_z$ ). Он возвращает спины–протоны в фазу (рефазировает их) и ведет к тому, что по истечении еще времени  $T_E/2$  появится достаточно сильный (регистрируемый) эхо-сигнал  $S(t)$ . Здесь через  $T_E$  (time echo) обозначено время регистрации (появления) эхо-сигнала. Обычно [79]  $T_E \sim 0.01 \div 0.1$  сек.

Изложенный вариант метода можно наглядно пояснить, используя *аналогию* с бегунами на дорожке [68, с. 127], [97, с. 44], [102, с. 43]. Стартер дает старт выстрелом ( $90^\circ$ -ный импульс) и бегуны (спины-протоны) разбегаются вдоль дорожки, но бегут с разными скоростями (расфазирование) и от их ног исходит слабый сигнал  $s(t)$ . Затем через  $t = T_E/2$  стартер дает второй выстрел ( $180^\circ$ -ный импульс), бегуны меняют направление своего движения на обратное, бегут со своими скоростями и "медленные" бегуны теперь оказываются впереди "быстрых". В результате при  $t = T_E$  все бегуны будут опять на старте и одновременно (в фазе) ударят ногами по дорожке — возникнет сильный эхо-сигнал  $S(t)$ .

Данная импульсная последовательность является *основной* при формировании ряда других последовательностей, например, спин-эхо-последовательности с множеством эхо-сигналов Хана (1950 г.) [68, с. 131].

Более конкретно, рассмотрим спин-эхо-последовательность импульсов Карра–Парселла (1954 г.). Последовательность импульсов  $90^\circ$ – $180^\circ$ , представленная на рис. 1.31, может давать значительные погрешности измерений из-за диффузии спинов, приводящей к тому, что ядра в случае неоднородности магнитного поля попадают в области поля с различной напряженностью. Для снижения влияния диффузии на измерения предназначена спин-эхо-последовательность *импульсов Карра–Парселла* (КП) [68], [157, Fig. 13]. В ней вслед за  $90^\circ$ -ным импульсом следует не один  $180^\circ$ -ный импульс (как в спин-эхо-ИП на рис. 1.31), а серия из нескольких  $180^\circ$ -ных импульсов. При этом интенсивности эхо-сигналов  $S_1(t)$ ,  $S_2(t)$ ,  $S_3(t)$ ,  $S_4(t)$ , ..., следующих за  $180^\circ$ -ными импульсами, спадают по закону  $\sim e^{-t/T_2}$  (см. рис. 1.32).

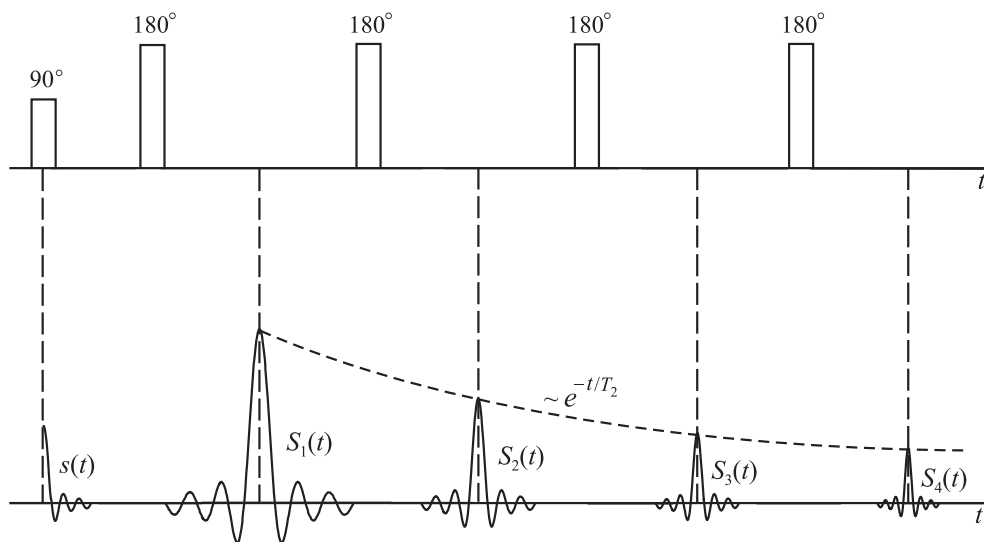


Рис. 1.32. Импульсная последовательность Карра–Парселла, ЯМР-сигнал  $s(t)$  и эхо-сигналы  $S_1(t)$ ,  $S_2(t)$ ,  $S_3(t)$ ,  $S_4(t)$ , ...

Такую ИП называют *мультиэхо* (multiecho) [97, с. 43]. Затем по прошествии времени повторения (repetition time)  $T_R$  может быть создан опять  $90^\circ$ -ный импульс и серия  $180^\circ$ -ных импульсов и т.д. [97, с. 44].

Существуют также следующие импульсные последовательности:

- ИП Мейбума–Джилла [1958 г.] — модификация последовательности Карра–Парселла, называемая, КПМД–последовательностью [68, с. 132–133],
- ИП насыщение–восстановление, или частичное насыщение, или  $90^\circ$ – $90^\circ$  [68, с. 122–124], [97, с. 37], [157, Fig. 10],
- ИП инверсия–восстановление, или  $180^\circ$ – $90^\circ$  [68, с. 124–125], [69, с. 61], [97, с. 39], [157, Fig. 11],
- ИП градиентное эхо [69, с. 62–63].

Для чего нужны последовательности импульсов? Они нужны для решения следующих задач.

1. Импульсы создают эхо-сигналы  $S(t)$ , а по эхо-сигналам можно решить основную задачу ЯМР-томографии – реконструкцию ЯМР-изображений (получение томограмм) (см. далее).

2. Частота ларморовой прецессии четко зависит от типа вещества (см. табл. 1.1), поэтому настраивая импульсы на разные частоты и получая эхо-сигналы на этих же частотах, мы будем получать томограммы для различных типов вещества.

3. С помощью ИП насыщение–восстановление вида  $90^\circ$ – $90^\circ$  можно определить времена продольной релаксации  $T_1$  [68, с. 129] для ряда органов и затем по значениям  $T_1$  можно поставить диагноз: здоровая ткань, предраковая ткань, рак (см. табл. 1.2).

**Градиентные поля.** Предположим, что исследуется, например, часть руки – локоть. На рис. 1.33 он изображен условно в виде цилиндра. Направим ось  $z$  вдоль оси цилиндра–локтя и отобразим в ряде  $z$ -сечений распределение плотности  $c_z(x, y)$ .

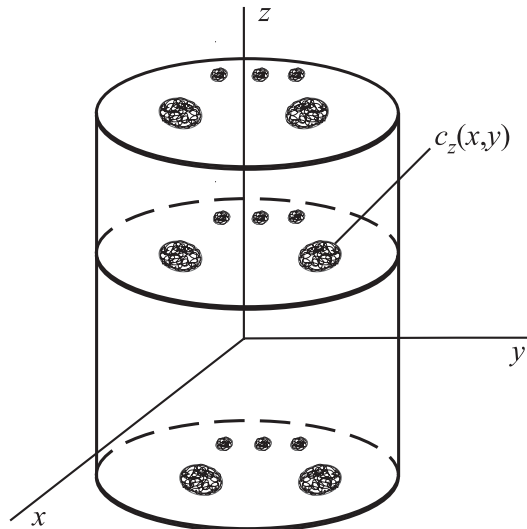


Рис. 1.33. Распределение плотности  $c_z(x, y)$  по  $z$ -сечениям

Пусть в объеме локтя создано постоянное магнитное поле  $B_0$ , направленное вдоль  $z$ . Временно будем полагать, что расфазирование спинов–протонов отсутствует и что



поле является однородным, т.е.  $\Delta B = 0$ . В этом случае все протоны будут прецессировать с одинаковой ларморовой частотой  $\omega = \gamma B_0$ . Если теперь создать на частоте  $\omega$   $\pi/2$ -импульс (рис. 1.28) или некоторую импульсную последовательность, например, спин-эхо-ИП (рис. 1.31), то мы получим ответный эхо-сигнал  $S(t)$  (рис. 1.31). Недостатком такого эксперимента является то, что этот эхо-сигнал будет идти от всего объема локтя на частоте  $\omega$ , а идея томографии состоит в том, что нужно получить эхо-сигнал отдельно от каждого  $z$ -сечения (или тонкого слоя) — тогда можно решать задачу реконструкции ЯМР-изображений (см. дальше). Отделить же различные сечения друг от друга можно, сопоставив им различные значения поля  $B$ , а значит, ларморовой частоты  $\omega$ . Тогда, принимая эхо-сигнал на различных частотах  $\omega$ , мы получим информацию отдельно для каждого сечения.

Чтобы каждому значению  $z$  соответствовало свое и только свое значение  $B$ , функция  $B(z)$  должна быть монотонно возрастающей (или убывающей). Проще (математически и технически) полагать закон изменения поля  $B(z)$  *линейным* — этому соответствует понятие градиентного магнитного поля.

**Определение.** *Градиентным  $z$ -полем* называется поле, индукция (или напряженность) которого  $G(z)$  изменяется по линейному закону:

$$G_z \equiv G(z) = g_z z, \quad (1.47)$$

где  $g_z = \text{const}$  — некоторая постоянная величина, означающая:  $g_z = \partial G(z)/\partial z$ .

Аналогично вводятся *градиентные поля по  $x$  и по  $y$* :

$$G_x = g_x x, \quad G_y = g_y y. \quad (1.48)$$

Градиентные магнитные поля формируются с помощью так называемых *градиентных катушек* [68, с. 209].

**Пример.** Проиллюстрируем роль градиентного поля на следующем простом примере [12], [68, с. 136]. Две пробирки наполнены водой (рис. 1.34).

Сначала (рис. 1.34а) вокруг пробирок формируем постоянное магнитное поле  $B_0$ . Создаем короткий мощный широкополосный  $\pi/2$ -импульс (см. рис. 1.28). От всех протонов обеих пробирок будет исходить суммарный затухающий сигнал ССИ  $s(t)$  типа (1.43). Примем этот сигнал широкополосным детектором. Если теперь выполнить преобразование Фурье над сигналом  $s(t)$ , то его спектр  $|S(\omega)|$  (рис. 1.34б) будет очень узким (типа рис. 1.30) с максимумом при  $\omega = \omega_0 = \gamma B_0$ , причем тем более узким, чем медленнее затухание сигнала, т.е. чем больше время релаксации  $T_1$ .

Такой же спектр будет получен, если  $\pi/2$ -импульс будет узкополосным с центральной частотой  $\omega_0 = \gamma B_0$ .

Если же к полю  $B_0$  добавить градиентное поле  $G_z = g_z z$  (рис. 1.34в) и создать широкополосный  $\pi/2$ -импульс, то спектр исходящего от пробирок сигнала (рис. 1.34г) будет более сложным, а именно, он будет иметь два широких максимума с центральными частотами  $\omega_1 = \gamma (B_0 + g_z z_1)$  и  $\omega_2 = \gamma (B_0 + g_z z_2)$ , где  $z_1$  и  $z_2$  —  $z$ -координаты средин пробирок, причем тем шире эти максимумы, чем шире пробирки вдоль  $z$ .

Если, наконец, создать узкополосный  $\pi/2$ -импульс с частотой  $\omega$ , то (при наличии  $B_0$  и  $G_z$ ) будут возбуждаться и испускать сигнал ССИ лишь спины-протоны

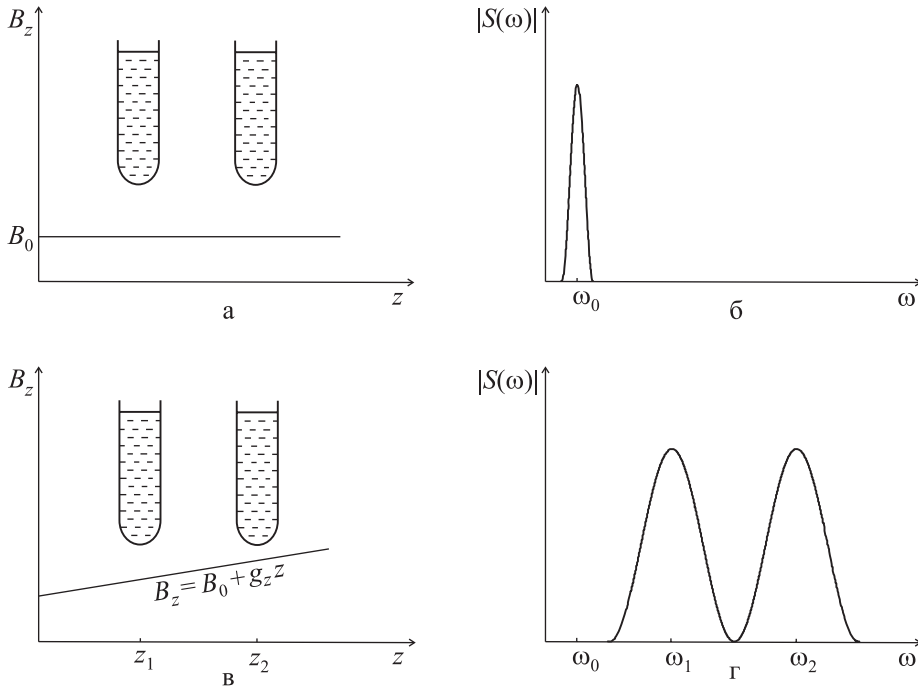


Рис. 1.34. Эксперименты с пробирками с водой, иллюстрирующие роль градиентного поля

с  $z$ -координатой  $z = (\omega - \omega_0) / \gamma g_z$  и, наоборот, задавая координату  $z$ -сечения, можно определить частоту  $\omega = \omega_0 + \gamma g_z z$ , на которой можно возбудить спины в этом сечении, чтобы они и только они испустили сигнал  $s_z(t)$ .

Теперь мы можем обратиться к основному вопросу – реконструкции ЯМР-изображений.

**Реконструкция ЯМР-изображений (основные принципы).** Под реконструкцией (формированием) ЯМР-изображений подразумевается определение распределения плотности протонов  $c_z(x, y)$  в  $z$ -сечениях по измеренным на разных частотах  $\omega = \omega(z)$  эхо-сигналам  $S_{\omega(z)}(t)$  [47–49, 68, 102, 139, 147].

"Сырые" (исходные, необработанные) данные хранятся в компьютерной матрице, называемой *к-пространством*. Термин "к-пространство" заимствован из математики. Букву "к" математики и физики используют для описания *частоты* света, звука и других волновых процессов [69, с. 56], [97, с. 83].

Основной принцип реконструкции ЯМР-изображений заключается в так называемом *пространственном кодировании частоты резонанса* [12, 30, 68, 69, 79, 97, 102, 139, 141, 147, 157]. Он состоит в том, что на однородное магнитное поле  $B_0$  накладываются градиентные поля, индукции которых зависят от координат  $x, y, z$ , а значит, и, наоборот, каждой точке  $(x, y, z)$  соответствует определенная индукция, а следовательно, частота и/или фаза. Другими словами, точка  $(x, y, z)$  пространства кодируется под частоту и/или фазу и, наоборот, под частотой и/или фазой закодированы координаты  $x, y, z$ .

Предположим, что включено однородное поле  $B_0$ , а также три градиентных поля  $G_x$ ,  $G_y$  и  $G_z$ . Тогда в некоторой точке  $(x, y, z)$  суммарное поле будет равно  $B_0 + G_x x + G_y y + G_z z$ , а резонансная частота (частота ларморовой прецессии) будет равна  $\omega(x, y, z) = \gamma (B_0 + g_x x + g_y y + g_z z)$ . Введение градиентных полей позволяет решать обратную задачу – определить координаты  $x, y, z$  точки, из которой исходит сигнал некоторой частоты  $\omega$ . Это – основа всех методик реконструкции ЯМР-изображений.

Множество точек, соответствующих некоторому значению  $\omega$ , образует в пространстве *плоскость* (если отвлечься от расфазирования) (рис. 1.35а):

$$\gamma (B_0 + g_x x + g_y y + g_z z) = \omega. \quad (1.49)$$

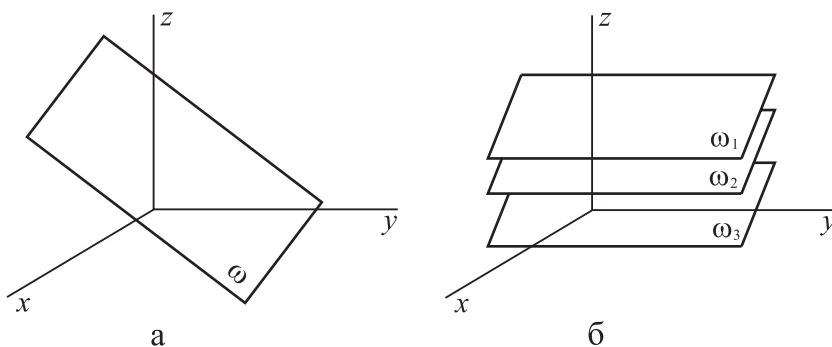


Рис. 1.35. а – плоскость постоянного значения  $\omega$  при произвольных  $g_x, g_y, g_z$ , б – плоскости постоянных  $\omega$  при  $g_z \gg g_x, g_z \gg g_y$

Обычно полагается, что  $g_z \gg g_x$  и  $g_z \gg g_y$ . Это создает так называемую *селекцию по  $z$* , в результате которой, во-первых, плоскости (1.49) получаются практически перпендикулярными оси  $z$ , а во-вторых, получается четкое разграничение по частотам  $\omega$  вдоль  $z$  (рис. 1.35б). Поэтому, принимая сигнал  $S(t)$  на некоторой частоте  $\omega$ , мы принимаем его практически из некоторого  $z$ -сечения, т.е.  $S(t) = S_{\omega(z)}(t)$ .

Если измерения выполнить на некоторой частоте  $\omega$ , то измеренный сигнал будет равен

$$S_{\omega(z)}(t) = A \iiint c(x, y, z) e^{i\gamma(B_0 + g_x x + g_y y + g_z z)t} dx dy dz, \quad (1.50)$$

где  $A$  – некоторая константа. Интегрирование в (1.50) ведется по плоскости (1.49) при производных  $g_x, g_y, g_z$  (рис. 1.35а). Если же  $g_z \gg g_x$  и  $g_z \gg g_y$ , то

$$S_{\omega(z)}(t) = A \iint c_z(x, y) e^{i\gamma(B_0 + g_x x + g_y y)t} dx dy \quad (1.51)$$

и интегрирование ведется по одной из плоскостей, практически перпендикулярной  $z$  (рис. 1.35б), причем множитель, содержащий  $g_z z$ , включен в  $A$ .

Заметим, что поскольку  $S(t)$  – волновой процесс, то при интегрировании введен экспоненциальный множитель  $\exp[i\gamma(\dots)t]$ , учитывающий фазы интегрируемых точек  $(x, y)$ .

Заметим также, что выражение (1.51) (также как и (1.50)) нельзя, строго говоря, рассматривать как интегральное уравнение относительно  $c_z(x, y)$ , так как исконая функция  $c_z(x, y)$  — двухмерная функция (при фиксированном  $z$ ), а измеренная функция  $S_{\omega(z)}(t)$  — лишь одномерная. Ниже будет показано, как сделать функцию  $S$  двухмерной, чтобы устранить это противоречие.

Прежде чем излагать конкретные схемы реконструкции ЯМР-изображений, рассмотрим подробнее идею селекции по  $z$ .

**Сечение и слой.** Предположим, что сформировано градиентное поле по  $z$ , а также однородное поле  $B_0$ , в результате суммарное поле вдоль  $z$  (без учета  $G_x$  и  $G_y$ ) будет равно

$$B_z \equiv B(z) = B_0 + g_z(z - z_0), \quad (1.52)$$

где  $z_0$  — некоторая константа. Если теперь возбуждать ансамбль протонов импульсом с некоторой частотой  $\omega$ , то будут возбуждаться протоны лишь в плоскости (сечении) с координатой

$$z = z_0 + (\omega - \gamma B_0) / \gamma g_z. \quad (1.53)$$

Однако исходящий из этой плоскости ЯМР-сигнал  $s(t)$  будет бесконечно слаб из-за бесконечно малой толщины сечения. Поэтому нужно возбуждать не плоскость, а *слой* некоторой эффективной толщины, которая должна быть достаточно "большой", чтобы получить сильный сигнал, и достаточно "малой", чтобы частоты протонов в пределах слоя не сильно отличались. Для этого импульс должен иметь не одну частоту, а набор частот.

Обычно тонкая структура импульса  $p(t)$  имеет следующий вид [30] (см. рис. 1.36).

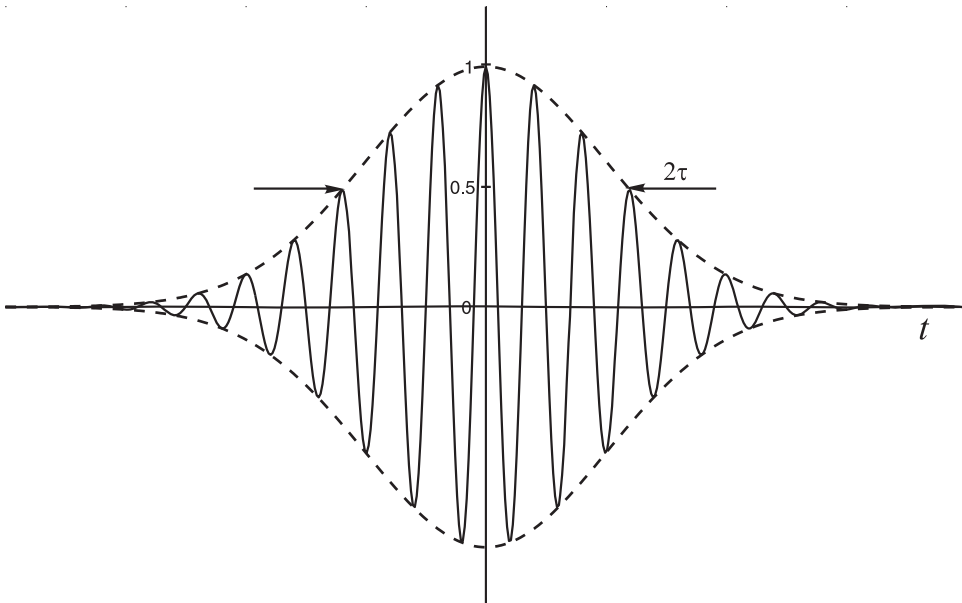


Рис. 1.36. Импульс  $p(t)$  (тонкая структура)

Он хорошо описывается формулой (с единичной амплитудой):

$$p(t) = \cos(\omega_0 t) \exp\left(-\frac{t^2}{2\sigma^2}\right), \quad (1.54)$$

т.е. представляет собой "косинус в гауссиане". Здесь  $\omega_0$  — несущая частота косинуса (а значит, импульса), а  $\sigma^2$  — дисперсия гауссианы, являющейся огибающей импульса.

Найдем полуширину по полувысоте  $\tau$  огибающей (гауссианы) — ее эффективную полуширину:

$$\exp\left(-\frac{\tau^2}{2\sigma^2}\right) = 0.5 = \exp(\ln 0.5) = \exp(-\ln 2),$$

откуда

$$\frac{\tau^2}{2\sigma^2} = \ln 2, \quad \tau = \sqrt{2 \ln 2} \sigma \approx 1.18 \sigma, \quad \sigma = \frac{\tau}{\sqrt{2 \ln 2}} \approx 0.85 \tau.$$

Теперь найдем преобразование Фурье (спектр) импульса:

$$\begin{aligned} P(\omega) &= \int_{-\infty}^{\infty} p(t) e^{i\omega t} dt = 2 \int_0^{\infty} e^{-t^2/2\sigma^2} \cos(\omega_0 t) \cos(\omega t) dt = \\ &= \exp\left(-\frac{\sigma^2}{2}(\omega - \omega_0)^2\right) + \exp\left(-\frac{\sigma^2}{2}(\omega + \omega_0)^2\right). \end{aligned} \quad (1.55)$$

Отбросим второе слагаемое в (1.55), получим окончательно:

$$P(\omega) = \exp\left(-\frac{\sigma^2}{2}(\omega - \omega_0)^2\right). \quad (1.56)$$

Видим, что спектр импульса есть гауссиана с центральной частотой  $\omega_0$ , причем спектр тем шире, чем уже импульс  $p(t)$  (см. рис. 1.37).

Обозначим через  $\Delta$  полуширину  $P(\omega)$  по полувысоте. Найдем связь между  $\Delta$  и  $\tau$ :

$$\exp\left(-\frac{\sigma^2}{2}\Delta^2\right) = 0.5 = \exp(-\ln 2), \quad \frac{\sigma^2}{2}\Delta^2 = \ln 2, \quad \Delta = \frac{\sqrt{2 \ln 2}}{\sigma} = \frac{2 \ln 2}{\tau},$$

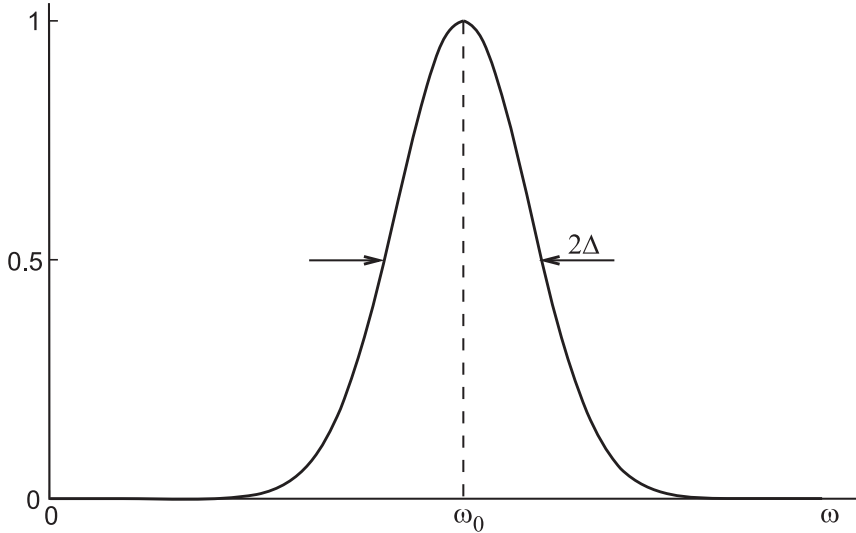
откуда полуширина спектра  $P(\omega)$  по полувысоте равна

$$\Delta = \frac{2 \ln 2}{\tau} \approx \frac{1.39}{\tau}. \quad (1.57)$$

Если включить градиентное поле (1.52), то в слое с центральной координатой  $z_0$  частота ларморовой прецессии будет изменяться как

$$\omega(z) = \gamma B_0 + \gamma g_z(z - z_0). \quad (1.58)$$

Теперь будем облучать вещество импульсом (1.54), причем положим  $\omega_0 = \gamma B_0$ . В этом случае будут возбуждаться в основном протоны в слое с центральной координатой  $z_0$  и с некоторой эффективной полутолщиной  $\delta z$ . Если в (1.58) обозначить

Рис. 1.37. Спектр  $P(\omega)$  импульса  $p(t)$ 

$\delta z = z - z_0$ ,  $\delta\omega = \omega(z) - \omega_0$  и приравнять  $\delta\omega = \Delta$  (см. (1.57)), то эффективная полутолщина слоя будет равна

$$\delta z = \frac{\delta\omega}{\gamma g_z} = \frac{2 \ln 2}{\gamma g_z \tau} = \frac{\sqrt{2 \ln 2}}{\gamma g_z \sigma}. \quad (1.59)$$

Если же, вообще говоря,  $\omega_0 \neq \gamma B_0$ , то меняя среднюю (несущую) частоту импульса  $\omega_0$ , можно изменять положение возбуждаемого слоя (его среднюю  $z$ -координату)

$$\bar{z} = z_0 + (\omega_0 - \gamma B_0) / \gamma g_z \quad (1.60)$$

(см. (1.53)). А меняя эффективную полуширину импульса  $\tau$  (или  $\sigma$ ), можно изменять эффективную полутолщину слоя  $\delta z$  (см. (1.59) при  $\omega_0 = \gamma B_0$ ).

**Методика Хоулта.** В пределах слоя эффективной толщины  $2\delta z$ , однако, протоны будут иметь различные ларморовы частоты (1.58) и будет иметь место расфазирование вдоль  $z$  (в отличие от расфазирования в плоскости  $x, y$ , изображенного на рис. 1.25), в результате чего суммарный эхо-сигнал от слоя будет очень слабым.

Для устранения этого эффекта Хоулт (1977 г.) предложил после действия  $\pi/2$ -импульса изменять направление градиентного поля  $G_z$  на противоположное, что ведет к рефазированию протонов и к появлению сильного эхо-сигнала  $S(t)$  (см. рис. 1.38) [30], [102, с. 48], [147, р. 272].

Методика Хоулта напоминает действие  $180^\circ$ -ного импульса (рис. 1.31, 1.32).

**Конкретные методы и схемы реконструкции ЯМР-изображений.** Исторически один из первых методов — *метод Лаутербура* (1972 г.) [161], названный им "зоигматографией" от греческого слова  $\zeta\omicron\iota\gamma\mu\alpha$  ("то, что объединяет") [12]. Затем (в 1975 г.) Кумар, Велти и Эрнст [159] разработали метод, названный *двухмерной фурье-зоигматографией* (Fourier zeugmatography) [12], [68, с. 138], [139], [147, р. 102].

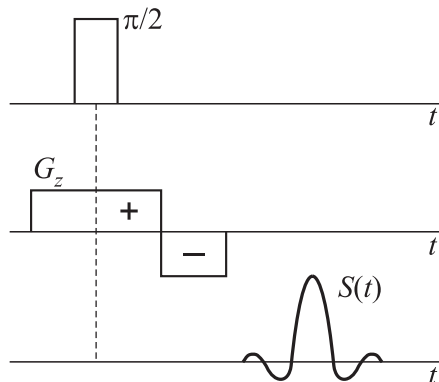


Рис. 1.38. Иллюстрация методики Хоулта

В дальнейшем термин "метод зойгмаграфии" был заменен на термин "метод реконструкции ЯМР-изображений" или "метод ЯМР-томографии".

В методе Кумара–Велти–Эрнста (см. рис. 1.39) для  $z$ -селекции создается градиентное поле  $G_z$  (с использованием методики Хоулта).

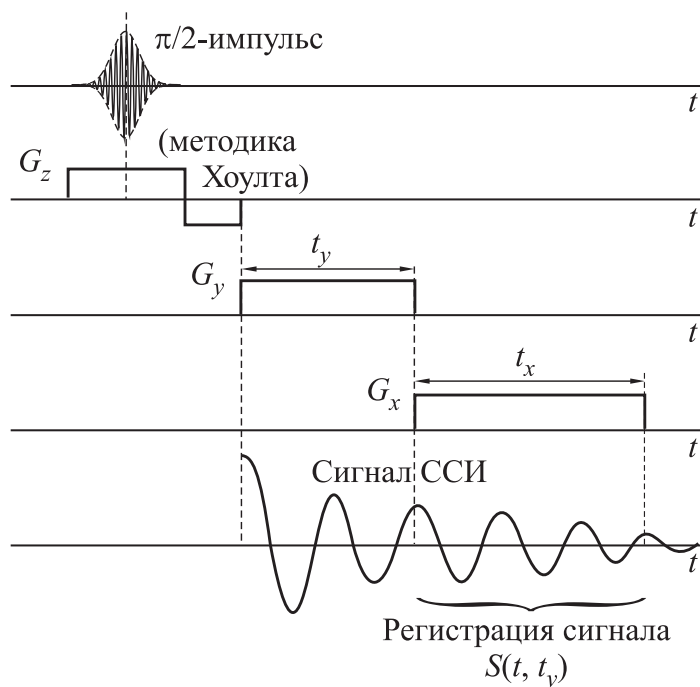


Рис. 1.39. Схема метода Кумара–Велти–Эрнста

Параллельно этому включается  $\pi/2$ -импульс для возбуждения спинов в образце. Затем включается градиентное поле  $G_y$  продолжительностью  $t_y$ , после чего включается градиентное поле  $G_x$  продолжительностью  $t_x$  и в течение времени  $t_x$  регистрируется сигнал ССИ  $S$ . Потом эксперимент повторяется: включаются  $G_y$  большей

продолжительности  $t_y$  и  $G_x$  той же продолжительности  $t_x$  и регистрируется новый сигнал  $S$  и т.д. Всего эксперимент выполняется  $n_y$  раз, что дает  $n_y$  сигналов ССИ в виде двухмерной функции  $S(t, t_y)$ .

Сигнал ССИ  $S(t, t_y)$  может быть записан как (ср. (1.51)) [147, р. 102]

$$S(t, t_y) = A \iint_{-\infty}^{\infty} c(x, y) e^{i\gamma(g_x x t + g_y y t_y)} dx dy. \quad (1.61)$$

Выражение (1.61) можно рассматривать как интегральное уравнение относительно  $c(x, y)$ . Оно весьма напоминает 2-мерное преобразование Фурье (ПФ). Чтобы оно действительно стало таковым, введем новые переменные:  $\omega_x = \gamma g_x x$ ,  $\omega_y = \gamma g_y y$ . Получим:

$$S(t, t_y) = A' \iint_{-\infty}^{\infty} c\left(\frac{\omega_x}{\gamma g_x}, \frac{\omega_y}{\gamma g_y}\right) e^{i(\omega_x t + \omega_y t_y)} d\omega_x d\omega_y, \quad (1.62)$$

где  $A' = A/\gamma^2 g_x g_y$ . Выражение (1.62) есть 2-мерное ПФ с масштабированием (scaling) [147, р. 24]. Из него с помощью обратного ПФ (с учетом масштабирования) можно получить:

$$c\left(\frac{\omega_x}{\gamma g_x}, \frac{\omega_y}{\gamma g_y}\right) = \frac{1}{2\pi A'} \iint_{-\infty}^{\infty} S(t, t_y) e^{-i(\omega_x t + \omega_y t_y)} dt dt_y. \quad (1.63)$$

Другими словами, сначала вычисляем обратное ПФ:

$$f(\omega_x, \omega_y) = \frac{1}{2\pi} \iint_{-\infty}^{\infty} S(t, t_y) e^{-i(\omega_x t + \omega_y t_y)} dt dt_y, \quad (1.64)$$

а затем — и искомую плотность:

$$c(x, y) = k f(\gamma g_x x, \gamma g_y y), \quad (1.65)$$

где  $k = 1/A' = \gamma^2 g_x g_y / A$ .

Наряду с методом Кумара–Велти–Эрнста для реконструкции ЯМР-изображений разработаны следующие методы:

- схема амплитудной вариации  $g_x$  [147, р. 272], или 1-я практическая схема [49], [102, с. 48] — в этой схеме используются  $\pi/2$ - и  $\pi$ -импульсы и градиентные поля  $G_z$ ,  $G_x$  и  $G_y$ , причем амплитуда  $g_x$  принимает ряд значений, в результате чего регистрируется сигнал ССИ в виде двухмерной функции  $S(t, g_x)$ ;

- схема вариации продолжительности  $G_x$ -поля [147, р. 272], или 2-я практическая схема [102, с. 50] — в этой схеме также используются  $\pi/2$ - и  $\pi$ -импульсы и градиентные поля  $G_z$ ,  $G_x$  и  $G_y$ , причем варьируется  $t_x$  — продолжительность  $G_x$ -поля, в результате чего регистрируется сигнал  $S(t, t_x)$  и данная схема становится весьма похожей на метод Кумара–Велти–Эрнста.



Существует еще ряд методов и схем обработки 2-мерных эхо-сигналов [12], [68, с. 147]: методы быстрой визуализации, например, эхо-планарный метод Мэнсфилда [162] и др.

Отметим при этом, что обработку указанными методами нужно выполнять на ряде частот  $\omega_z$  — этому будет соответствовать определение плотности  $c(x, y)$  в ряде сечений с различными  $z$ , т.е.  $c_z(x, y)$ . Наряду с этим разработаны так называемые интегральные методы, например, метод 3-мерной зоигматографии [12], [147, р. 274], в котором варьируются значения  $t_x$ ,  $t_y$  и  $t_z$ , а 3-мерное изображение  $c(x, y, z)$  реконструируется с помощью 3-мерного ПФ сигнала ССИ  $S(t_x, t_y, t_z)$ .

**Влияние неоднородности полей на разрешающую способность ЯМР-изображений.** Пусть создано поляризирующее поле (рассмотрим его зависимость от координаты  $x$ ) [49]:  $B(x) = B_0 + \Delta B_0(x)$ , где  $B_0 = \text{const}$ , а  $\Delta B_0(x)$  — неоднородность поля, а также градиентное поле  $G_x(x) = g_x x$ . Тогда, как показано в [30], неоднородность  $\Delta B_0(x)$  ведет к тому, что в ЯМР-изображении (томограмме) будут разрешаться детали с расстоянием между ними

$$\delta x \geq \frac{B_0}{g_x} \frac{\Delta B_0(x)}{B_0}. \quad (1.66)$$

Видим, что разрешающая способность томограмм понижается ( $\delta x$  увеличивается) пропорционально относительной неоднородности поляризирующего поля  $\Delta B_0(x)/B_0$  и обратно пропорционально  $g_x$ .

Можно также записать оценку относительной неоднородности поляризирующего поля, обеспечивающей заданное разрешение  $\delta x$ :

$$\frac{\Delta B_0(x)}{B_0} \leq \frac{g_x}{B_0} \delta x. \quad (1.67)$$

Если же созданы поля  $B(x) = B_0$  и  $G_x(x) = g_x x + \Delta g_x(x)x$ , где  $\Delta g_x(x)$  — отклонение от линейности градиентного  $x$ -поля, то относительное отклонение от линейности градиентного  $x$ -поля, соответствующее  $\delta x$ , выражается неравенством:

$$\frac{\Delta g_x(x)}{g_x} \leq \frac{\delta x}{x}. \quad (1.68)$$

Аналогичны оценки для  $\Delta B_0(y)$ ,  $\Delta B_0(z)$ ,  $\Delta g_y(y)$ ,  $\Delta g_z(z)$ .

**П р и м е р .** Пусть  $B_0 = 0.4$  Тл,  $g_x = 5$  мТл/м, диаметр рабочего объема ЯМР-томографа  $x = 20$  см, требуемая разрешающая способность  $\delta x = 0.2$  мм. Тогда

$$\frac{\Delta B_0}{B_0} \leq 2.5 \cdot 10^{-6}, \quad \frac{\Delta g_x(x)}{g_x} \leq 10^{-3}, \quad (1.69)$$

т.е. для обеспечения высокой разрешающей способности томограмм (в доли мм) необходима высокая степень однородности поляризирующего поля  $B_0$  (порядка  $10^{-6}$ ) и линейности градиентных полей  $G_x$ ,  $G_y$ ,  $G_z$  (порядка  $10^{-3}$ ).

**О п р е д е л е н и е.** *Рабочим объемом* называется та область зоны доступа томографа (точнее, катушки магнита томографа), в которой выполняются требования по разрешающей способности и относительным отклонениям полей типа (1.66)–(1.69).

В оценке (1.68)  $x$  — линейный размер (диаметр) рабочего объема. Обычно диаметр рабочего объема не превышает 40–50 % диаметра зоны доступа.

Вопросы создания высокородных магнитных полей с помощью различных магнитов (сверхпроводящих, резистивных и постоянных) изложены во многих публикациях ([31, 75, 80] и др.).

**Области применения ЯМР-томографии.** Основная область применения МРТ, как и РКТ, — *медицина* [68, 69, 97, 147], где различные методы МРТ позволяют выполнять эффективные обследования мозга пациента, его легкого, кишечника, желудка, печени, селезенки и т.д. с целью диагностики различных заболеваний и патологий: выявление предраковых опухолей и метастаз на ранней стадии их развития, обследование различных заболеваний мозга (болезни Ходжкина, Альцгеймера, глиома, рассеянный склероз) и т.д.

МРТ применяется также в *биологии* (исследование биологических жидкостей, клеток и т.д.), в *материаловедении* (исследование пластмасс, стекол, керамики, полимеров и т.д.), в *пищевой промышленности* (анализ качества продуктов, вин, масел и т.д.). Важным разделом МРТ является *магнитно-резонансная спектроскопия* (МРС) [68, 79], позволяющая с высокой точностью измерять спектры различных химических элементов.

Для изучения веществ на молекулярном уровне используется *ЯМР-микроскопия*, позволяющая обнаруживать микродефекты в пластмассовых трубах, находить алмазы в породе, обследовать биологические микрообъекты (вирусы, белки, аминокислоты и т.д.).

**Примеры реконструкции ЯМР-изображений.** На рис. 1.40 в качестве примера представлены две ЯМР-томограммы.

### Контрольные задания и вопросы

1. В чем заключается эффект ЯМР (изложить суть открытия Штерна–Герлаха, Уленбека–Гаудсмита, Раби, Блоха–Парселла и Завойского; объяснить понятия: спин, магнитный момент, прецессия, гиромангнитное отношение, частота ларморовой прецессии).

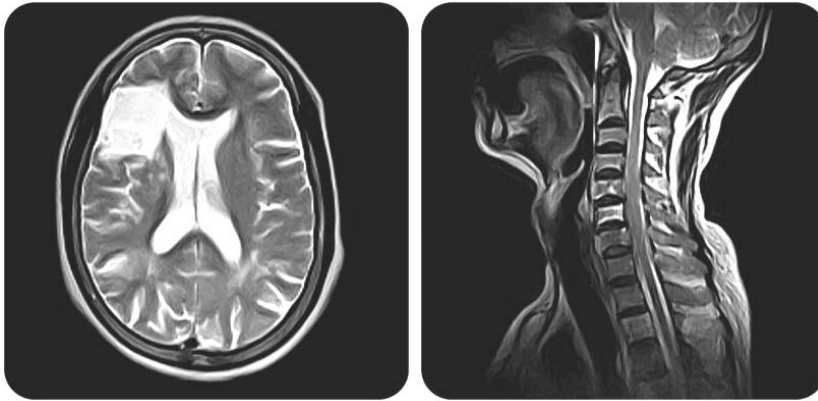
2. Объяснить на основе уравнения Лармора характер движения магнитного момента протона в постоянном магнитном поле.

3. Что такое ансамбль протонов и вектор ядерной намагниченности ансамбля протонов?

4. Опишите движение магнитного момента протона в постоянном и переменном магнитных полях (осцилляция, прецессия, нутация, частота вращения полей, расстройка по частоте, резонанс).

5. Запишите уравнения Блоха и изложите физический смысл слагаемых, входящих в уравнения (прецессия, релаксация, локальные поля, расфазирование).

6. Выведите решения дифференциальных уравнений Блоха путем их интегрирования (задание повышенной трудности).



а

б

Рис. 1.40. Две ЯМР-томограммы.

а — сечение головного мозга,

б — сечение в районе шейного отдела позвоночника

7. Что такое ЯМР-сигнал, или сигнал ССИ? Условие его появления и причина его затухания?
8. Запишите закон убывания ЯМР-сигнала  $s(t)$  и выведите формулы для его спектра  $S(\omega)$ .
9. Изложите действие импульсной последовательности для регистрации спин-эха (или спин-эхо-ИП). Приведите аналогию с бегунами.
10. Изложите действие импульсной последовательности Карра–Парселла.
11. Для чего нужны последовательности импульсов?
12. Для чего вводятся градиентные поля по  $z$ ,  $x$  и  $y$ ? Проиллюстрируйте роль градиентного поля на примере двух пробирок с водой.
13. Изложите основные принципы реконструкции ЯМР-изображений (пространственное кодирование частоты резонанса, селекция по  $z$ , интегральное выражение для измеренного сигнала  $S(t)$ ).
14. Сечение, слой, тонкая структура импульса  $p(t)$  и его спектр  $P(\omega)$ . В чем состоит связь этих понятий?
15. В чем суть методики Хоулта?
16. Изложите метод двумерной фурье-зоигматографии Кумара–Велти–Эрнста. Что является самым существенным в этом методе?
17. Как влияет неоднородность полей на разрешающую способность ЯМР-изображений (томограмм)? Приведите пример.
18. В каких областях применяется МРТ (и МРС)?

## Глава 2

# РЕКОНСТРУКЦИЯ ИСКАЖЕННЫХ ИЗОБРАЖЕНИЙ

В этой главе рассмотрим задачу обработки искаженных изображений [6, 18, 27, 35, 36, 39, 40, 45, 46, 52, 91, 92, 102, 120, 129, 137, 142, 143, 145, 172].

### 2.1. Объекты изображений, системы наблюдений, типы искажений

**Объекты изображений.** Под изображениями будем подразумевать фотоснимок человека, текста, природы, телескопический снимок космического объекта (планеты, туманности, звездного скопления, галактики), снимок земной поверхности со спутника, живописную картину, биологические микроорганизмы (вирусы, ДНК), наблюдаемые в микроскоп, расплавленный металл, оптико-электронное изображение объекта, изображение движущейся цели (самолета, автомобиля, танка), томограмму, номерной знак автомобиля и т.д.

**Системы наблюдений.** Под *системами* наблюдений, измерений и обработки изображений будем подразумевать системы, включающие устройства регистрации изображений (фотоаппараты, цифровые фото- или видеокамеры, микроскопы, телескопы, томографы и т.д.), а также вычислительные устройства (ПК, СВУ и др.).

Полагаем, что изображение регистрируется матрицей сенсоров на основе ПЗС-матрицы (сенсор — это чувствительный элемент, преобразующий освещенность в напряжение [35, с. 90], а ПЗС — прибор с зарядовой связью [19, с. 96], [35, с. 95], [133], [142, с. 33]). При этом выполняются оцифровка, дискретизация и квантование изображения [35, 129, 142, 143]. В результате изображение будет представлено матрицей интенсивностей из  $m$  строк и  $n$  столбцов или  $m \times n$  пикселей (будем писать кратко: пкс) [35, с. 94–102]. *Пиксель* (pixel) — это точка на дискретной  $2D$ -сетке (сокращение от picture element — элемент изображения) [142, с. 41]. Устройством, выполняющим операции оцифровки, дискретизации и квантования, может быть цифровая фото- или

видеокамера, однако будем относить к рассматриваемым устройствам регистрации изображений (УРИ) и фотоаппарат с пленкой при условии, что выполнены дополнительно оцифровка, дискретизация и квантование изображения–фотоснимка.

**Типы искажений изображений.** При наблюдении и измерении изображений объектов могут иметь место *искажения*. Можно выделить следующие основные *типы искажений изображений* [6], [142, с. 27]:

1. Искажения, не требующие для их устранения математической обработки: царапины на снимке, неудачно подобранная яркость и контрастность.

2. Искажения, требующие примитивной математической обработки: геометрические искажения (нужно изменить масштаб по горизонтали или вертикали или устранить нелинейность).

3. Физико-технические искажения: абберации — искажения, формируемые оптической системой (линзой фотоаппарата или зеркалом телескопа) в силу ее "неправильной" формы (см. далее п. 2.2).

4. Дифракция света на диафрагмах и оправках линз, обусловленная волновой природой света.

5. Искажения, требующие сложной математической обработки (смаз, дефокусировка, зашумленность изображения; см. далее п. 2.3–2.5).

Наука об обработке изображений имеет специальное название — *иконика* [52, 129]. Слово *иконика* происходит от греческого слова *εικον* — изображение, образ. В настоящее время под термином "иконика" подразумевают компьютерную обработку изображений.

Обработка изображений обычно включает [6, с. 13] распознавание образов, кодирование и непосредственную обработку изображений (устранение искажений вышеперечисленных пяти типов). Мы не будем рассматривать распознавание образов и кодирование изображений, а что касается устранения искажений, то рассмотрим лишь искажения типов 1 и 3 (кратко) и типа 5 (подробно).

Отметим при этом, что в работе [6] делается различие между восстановлением и реконструкцией изображений: под *восстановлением изображений* подразумевается оценка параметров искажения (например, величины смаза  $\Delta$ , см. (3)) и использование их для коррекции изображений, а под *реконструкцией* изображений подразумевается определение деталей в сильно искаженных изображениях. Отметим также, что в работе [142, с. 27] под *восстановлением* изображений подразумевается устранение расфокусировки, размытости, шумов и т.д., а под *реконструкцией* — получение изображений с помощью непрямых методов (томографических и др.) — это примыкает к понятию задач второго типа (см. Введение). Мы не будем делать четкого различия между терминами "восстановление" и "реконструкция", тем не менее, в обработке изображений предпочтение будем отдавать термину "восстановление", а в томографии — термину "реконструкция" (см. гл. 1).

**Типы изображений и классы чисел в системе MatLab.** Рассмотрим представление изображений в системе MatLab в пакете Image Processing Toolbox (IPT) [36, 44–46, 142]. При компьютерной реализации изображения могут быть следующих типов [36, с. 40], [45, с. 470]:

- *бинарные* (черно-белые, двоичные, logical, bw), используемые иногда для отображения (но не для обработки) текстов;
- *полутоновые* (серые, gray, grayscale) — в каждом пикселе смесь красного (R), зеленого (G) и синего (B) цветов в одинаковой пропорции, в результате чего получаются оттенки серого цвета;
- *цветные* (полноцветные, RGB, rgb) — в каждом пикселе смесь R, G и B цветов в произвольной пропорции, что позволяет довольно точно имитировать цвета различных устройств цветовоспроизведения (цветные мониторы, цветные принтеры, глаза) [35, с. 427] — идея, идущая от М.В. Ломоносова;
- *индексированные* (цветные палитровые, ind) — разновидность RGB-изображений, использующая цветовую карту (map).

Отметим также некоторые, необходимые для обработки изображений, *классы чисел* в системе MatLab в пакете IPT [36, с. 40], [45, с. 475]:

- `uint8` — беззнаковое (unsigned) целое (int) 8-битовое число, принимающее значения  $0, 1, \dots, 255$  (поскольку  $2^8 = 256$ );
- `uint16` — беззнаковое целое 16-битовое число со значениями  $0, 1, \dots, 65535$ ;
- `double` — вещественное число с плавающей точкой двойной точности (16 цифр) в диапазоне от  $2^{-1022} \approx 10^{-308}$  до  $2^{1023} \approx 10^{308}$  (8 байт на число);
- `logical` — число со значениями 0 или 1 (1 байт на число).

**Соглашение о координатах.** В различных книгах по-разному обозначают координаты. Например, в [35, с. 102], [142, с. 42] начальные индексы массивов, в частности, матрицы  $a$  имеют нулевые значения:  $a_{0,0}$ , а последние индексы — значения  $M - 1$  и  $N - 1$ :  $a_{M-1, N-1}$ , где  $M$  — число строк, а  $N$  — число столбцов матрицы  $a$ . В [35] вертикально вниз направлена ось  $x$  (и индекс  $i$  — номер строки), а горизонтально — ось  $y$  (и индекс  $j$  — номер столбца). В [142] ось  $x$  (и индекс  $n$ ) направлена горизонтально, а ось  $y$  (и индекс  $m$ ) — вертикально вниз и элемент матрицы записывается как  $a_{m,n}$ . Однако, если книга ориентирована на MatLab, то начальные индексы у массивов должны иметь единичные значения. В книге [36, с. 29, рис. 2.1(б)] именно так и сделано: верхний левый угол матрицы изображения имеет координаты  $(r, c) = (1, 1)$ , а нижний правый —  $(r, c) = (M, N)$ , причем ось  $x$  (и индекс  $r$  — номер строки) направлена вниз, а ось  $y$  (и индекс  $c$  — номер столбца) — горизонтально, при этом для обозначения номеров строки и столбца не используются символы  $i$  и  $j$ , поскольку в системе MatLab  $i$  и  $j$  обычно используются для обозначения мнимой единицы  $\sqrt{-1}$ .

Тем не менее, в данной книге (в гл. 2) мы будем использовать  $i$  для обозначения номера столбца, а  $j$  — для номера строки изображения, причем ось  $x$  (и номер  $i$ ) направим горизонтально слева направо, а ось  $y$  (и номер  $j$ ) — вертикально вниз, при этом  $i = 1, 2, \dots, N$ ,  $j = 1, 2, \dots, M$ . Будем обозначать интенсивность, например, на смазанном изображении в непрерывном случае через  $g_y(x)$ , а в дискретном случае — через  $g_j(i)$ , где  $x$  и  $i$  направлены вдоль смаза. Аналогично, будем обозначать интенсивность на дефокусированном изображении в непрерывном случае через  $g(x, y)$ , а в дискретном случае — через  $g(j, i)$ . При этом под дискретным случаем будем подразумевать представление изображения в виде набора пикселей. Разночтение в обозначениях:  $g(x, y)$  и  $g(j, i)$ , где  $x$  и  $i$  (а также  $y$  и  $j$ ) имеют одинаковое направле-

ние, обусловлено тем, что в непрерывном случае принято первым аргументом ставить горизонтальную ось, а в дискретном случае — номер строки  $j$ .

**Примеры искаженных изображений.** В качестве иллюстрации к вышеизложенному на рис. 2.1 и 2.2 приведены примеры искаженных изображений.

На рис. 2.1 приведено смоделированное смазанное под углом серое изображение текста (текст был набран в редакторе Word, скопирован в PaintBrush, сохранен как \*.bmp, прочтен в системе MatLab m-функцией `imread`, добавлен слабый фон — имитация бумаги и выполнено смазывание изображения m-функциями системы MatLab `fspecial` и `imfilter`, о которых речь пойдет дальше).

618x690

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}, \quad (2.1)$$

$$f_1 = 300, \quad f = 35,$$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.2)$$

$$f_2 = 354, \quad q = f_1 / f_2 = 84.7.$$

Рис. 2.1. Смазанное серое изображение текста.

Величина смаза  $\Delta = 25$  пкс, угол смазывания  $\theta = 30^\circ$ , фон = 5

На рис. 2.2 приведено дефокусированное и зашумленное серое изображение человека (файл `cameraman.tif`).



Рис. 2.2. Дефокусированное и зашумленное импульсным шумом изображение человека (фотографа)

Это изображение используется как фантом во многих публикациях ([102, 148, 150] и др.). Подробности о смазывании, дефокусировании и зашумленности см. ниже.

На цветной вклейке (рис. 2.72) приведено дефокусированное цветное (RGB) изображение мяча (`football.jpg`) и результат его реконструкции. Подробности обработки цветных изображений, а также примеры других искаженных изображений см. далее.

### Контрольные задания и вопросы

1. Перечислите возможные объекты изображений.
2. Перечислите системы наблюдений. Расскажите кратко о ПЗС-матрице и операциях оцифровки, дискретизации и квантования изображения.
3. Назовите типы искажений изображений. Что включает в себя обработка изображений?
4. Назовите типы изображений и классы чисел в системе MatLab.
5. Какие бывают соглашения о координатах? Какое соглашение о координатах  $i$ ,  $j$ ,  $x$ ,  $y$  принято в данной книге?

## 2.2. Аберрации оптических систем

Рассмотрим кратко вопрос об аберрациях оптических систем на примерах линз фотоаппаратов и зеркал телескопов [67, с. 276–291], [128, с. 7], [134].

**Типы аберраций.** "*Аберрация*" происходит от латинского слова *aberratio* — отклонение и означает искажения изображений, формируемые самими оптическими системами (линзами и зеркалами) вследствие их "неправильной" формы. Основные *типы аберраций* — геометрические и хроматическая. Геометрические аберрации бывают следующих видов: сферическая аберрация, кома, астигматизм, дисторсия, кривизна поля изображения. Все эти типы аберраций устраняются (или уменьшаются) непостоянством кривизны линз или зеркал и/или подбором дополнительных элементов. Вследствие волновой природы света возникают также искажения изображений в виде дифракции света на диафрагмах, оправках линз и т.д. Они обычно влияют на качество изображений меньше, чем геометрические и хроматическая аберрации.

**Сферическая аберрация.** Остановимся специально на сферической аберрации в связи с тем, что она имеет прямое отношение к космическому телескопу "Хаббл" (КТХ) [135, 173], а пример КТХ хорошо иллюстрирует возможности математической реконструкции изображений.

Суть сферической аберрации состоит в том, что световые лучи, идущие параллельно оптической оси осесимметричной линзы (или зеркала) постоянной кривизны вблизи оси и вдали от нее, не собираются в одну точку, а дают размытое пятно (см. рис. 2.3).

Рассмотрим параллельные лучи, исходящие от одной точки объекта, и линзу со сферической поверхностью постоянной кривизны (рис. 2.3а). Лучи, проходящие через центральную зону линзы (вблизи ее оптической оси), собираются почти все в так называемой гауссовой плоскости (ГП), а лучи, удаленные от оси, собираются ближе к линзе. В результате от каждой точки объекта в ГП будет формироваться не точка, а пятно (кружок рассеяния). Как следствие, все изображение в ГП будет нечетким (размытым). Если же экран помещать не в ГП, а ближе к линзе, то диаметр кружка



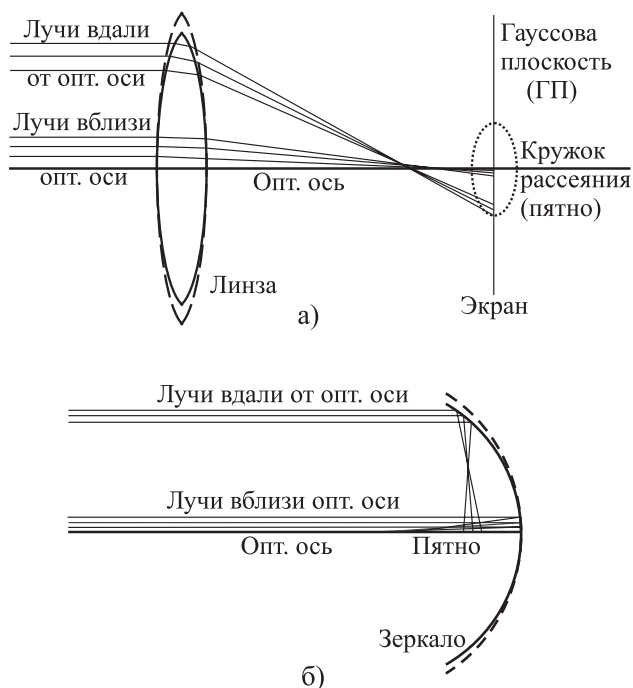


Рис. 2.3. Сферическая аберрация у линзы (а) и у зеркала (б)  
(пунктиром отмечены линза и зеркало,  
у которых устранена сферическая аберрация)

может стать в некотором положении примерно в 4 раза меньше, чем в ГП, но тем не менее, не близким к нулю.

Чтобы устранить сферическую аберрацию у линзы, нужно:

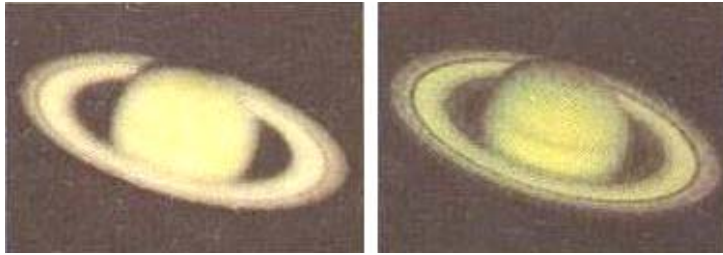
1) радиус кривизны линзы сделать непостоянным, увеличивая его к краям (пунктир на рис. 2.3а) или

2) использовать дополнительные элементы с асферическими поверхностями.

На рис. 2.3б изображена ситуация со сферической аберрацией, характерная для сферического зеркала постоянной кривизны, например, для зеркала телескопа.

**О космическом телескопе "Хаббл".** В июне 1990 г. в космос был запущен зеркальный космический телескоп "Хаббл" диаметром 2.4 м (по фамилии известного американского астронома Хаббла) [135, 173]. При изготовлении зеркала КТХ, чтобы устранить сферическую аберрацию, радиус кривизны зеркала был увеличен к краям (как отмечено пунктиром на рис. 2.3б). Однако изготовители "перестарались": края зеркала были сделаны на 2 мкм ( $\approx 1/15$  толщины человеческого волоса) более плоскими, чем это было нужно для устранения сферической аберрации. В результате у зеркала из-за излишней коррекции (overcorrection) возникла остаточная аберрация. Это, во-первых, привело к тому, что лишь 15% света от точечного источника (например, далекой звезды) концентрировалось в кружке радиусом  $0.1''$  вместо положенных 70%, а во-вторых, привело к дефокусированию изображений космических объектов.

В декабре 1993 г. сферическая аберрация зеркала КТХ была устранена астрономами (с помощью пяти пар малых корректирующих зеркал). А в промежутке продолжительностью 2.5 года (с июня 1990 г. по декабрь 1993 г.) многие из полученных дефокусированных изображений были откорректированы математико-компьютерным путем. На рис. 2.4 в качестве примера представлено два изображения Сатурна: слева — необработанное изображение, переданное с КТХ до его коррекции, а справа — исправленное путем математической обработки четкое изображение [135].



а

б

Рис. 2.4. Необработанное изображение Сатурна (а) и исправленное изображение (б)

### Контрольные задания и вопросы

1. Что такое аберрация оптической системы?
2. Перечислите основные типы аберраций.
3. Подробно изложите суть сферической аберрации на примере линзы и зеркала.
4. Расскажите о ситуации с космическим телескопом "Хаббл".

## 2.3. Реконструкция смазанных изображений (теория)

Рассмотрим данную задачу на примере смазанного (смещенного, сдвинутого) фотоснимка. Полагаем, что другие типы искажений (царапины, аберрации и т.д.) отсутствуют или устранены.

**Прямая задача.** Считаем фотографируемый объект плоским вследствие его удаленности. Пусть объект и фотопленка фотоаппарата расположены параллельно апертуре тонкой линзы фотоаппарата по разные стороны от линзы на расстояниях от нее  $f_1$  и  $f_2$  соответственно (рис. 2.5).

При этом

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}, \quad (2.1)$$

где  $f$  — фокусное расстояние линзы и  $f_1 \geq f$ .

Введем в плоскости объекта прямоугольную систему координат  $\xi'O'\eta'$ , а в плоскости фотопленки —  $\xi O\eta$ . Рассмотрим на объекте некоторую точку  $A'(\xi', \eta')$  с интенсивностью излучения  $w'(\xi', \eta')$ . Лучи, исходящие из нее и прошедшие через линзу

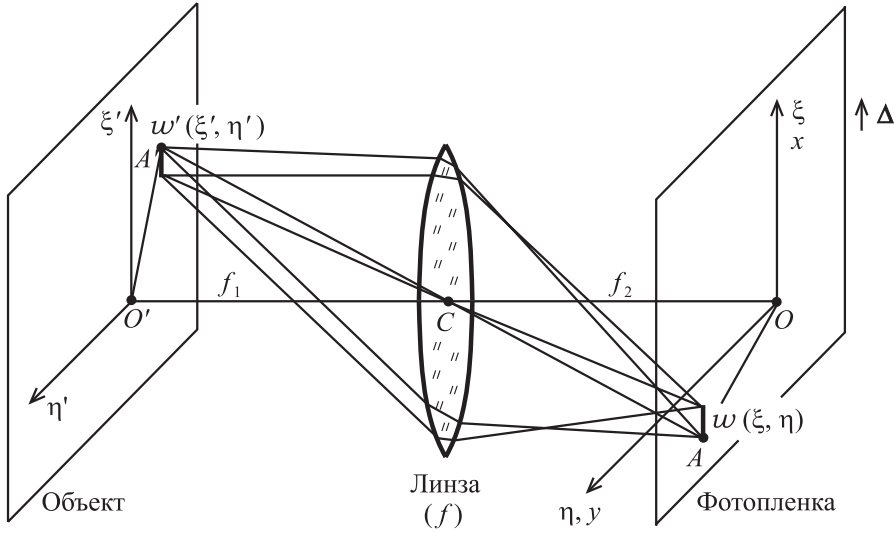


Рис. 2.5. Схема получения смазанного изображения

(сплошные прямые линии из точки  $A'$  на рис. 2.5), пересекутся в некоторой точке  $A(\xi, \eta)$  в плоскости фотопленки. Из подобия треугольников  $A'CO'$  и  $ACO$  имеем:

$$\frac{\overrightarrow{O'A'}}{f_1} = \frac{\overrightarrow{OA}}{f_2}.$$

В проекциях получим:

$$\frac{\xi'}{f_1} = -\frac{\xi}{f_2}, \quad \frac{\eta'}{f_1} = -\frac{\eta}{f_2}. \quad (2.2)$$

В результате точка  $A(\xi, \eta)$  на фотопленке будет иметь интенсивность  $w$  и координаты  $\xi, \eta$  (*прямая задача*):

$$w(\xi, \eta) = w'(\xi', \eta'), \quad \xi = -\xi'/q, \quad \eta = -\eta'/q, \quad (2.3)$$

где  $q = f_1/f_2$ , а  $f_2$  равняется

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}. \quad (2.4)$$

Таким образом, каждой точке  $A'$  на объекте будет соответствовать точка  $A$  на фотопленке с той же интенсивностью  $w(\xi, \eta) = w'(\xi', \eta')$ , но с измененными в  $q$  раз и перевернутыми координатами (см. (2.3)).

**Пример 1.** Пусть  $f_1 = 3$  м,  $f = 3.5$  см, тогда  $f_2 = 3.5413$  (согласно (2.4)), а  $q = 84.7143$ , т.е. изображение будет в 84.7 раз меньше объекта (и перевернуто).

**Пример 2.** Пусть  $f_1 = \infty$  (Солнце, звезда, Луна, планета), тогда  $f_2 = f$ , а  $q = \infty$ .

По фотоснимку можно восстановить объект (*обратная задача* в отсутствие смаза):

$$w'(\xi', \eta') = w(\xi, \eta), \quad \xi' = -q\xi, \quad \eta' = -q\eta, \quad f_1 = \left( \frac{1}{f} - \frac{1}{f_2} \right)^{-1}. \quad (2.5)$$

**Обратная задача устранения смаза.** Полагаем, что за время экспозиции  $\tau$  фотоаппарат с пленкой совершил прямолинейный и равномерный *сдвиг* (смещение) со скоростью  $v = \text{const}$  вдоль некоторого направления, т.е. на величину  $\Delta = v\tau$  или сдвиг совершил объект (например, движущаяся цель) на величину  $-q\Delta$ . Направим ось  $\xi$  вдоль сдвига. В результате изображение на фотопленке будет смазанным (сдвинутым, смещенным) вдоль  $\xi$  (см. рис. 2.6 [96]).

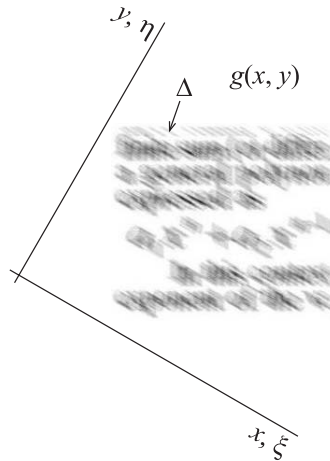


Рис. 2.6. Пример смазанного изображения.

$\Delta$  – сдвиг (смаз, смещение), ось  $\xi$  направлена вдоль смаза

**З а м е ч а н и е.** Об определении величины  $\Delta$  и направления сдвига (смаза) речь пойдет в п. 2.4.

Возникает *обратная задача* (при наличии смаза): восстановить (реконструировать) математическим путем истинное (неискаженное изображение), зная смазанное изображение, а также величину смаза  $\Delta$  и направление смаза.

Чтобы решить эту задачу, сделаем вначале математическое описание задачи.

**Вывод интегрального уравнения.** Введем наряду с неподвижной системой координат  $\xi O\eta$  подвижную систему координат  $xOy$  (ось  $x$  направим вдоль  $\xi$ ), связанную с движущейся фотопленкой и совпадающую с  $\xi O\eta$  лишь в начальный момент процесса смазывания (рис. 2.5 и 2.6). На некоторую точку  $(x, y)$  фотопленки за время экспозиции  $\tau$  спроектируется непрерывный набор точек  $A$  с абсциссами от  $\xi = x$  до  $\xi = x + \Delta$  и с различными интенсивностями  $w(\xi, y)$ . Результирующая интенсивность (обозначим ее через  $g$ ) в некоторой точке  $(x, y)$  фотопленки будет равна сумме (интегралу) интенсивностей  $w(\xi, y)$ ,  $\xi \in [x, x + \Delta]$ :

$$g(x, y) = \frac{1}{\Delta} \int_x^{x+\Delta} w(\xi, y) d\xi. \quad (2.6)$$

В (2.6) перед интегралом поставлен множитель  $1/\Delta$ , чтобы при  $\Delta \rightarrow 0$  (в отсутствие смаза) мы имели:  $g(x, y) \rightarrow w(x, y)$ , а также при  $w(\xi, y) = w = \text{const}$  выполнялось равенство:  $g(x, y) = w$  (ср. [102, с. 66]).

Запишем (2.6) иначе:

$$\frac{1}{\Delta} \int_x^{x+\Delta} w(\xi, y) d\xi = g(x, y). \quad (2.7)$$

Соотношение (2.7) является основным в задаче реконструкции смазанных изображений [2, 4, 18, 35, 96, 102, 109, 112, 120, 168]. В нем  $g(x, y)$  — измеренное распределение интенсивности на смазанном изображении (на фотопленке) в функции прямоугольных координат  $x, y$ , причем ось  $x$  направлена вдоль смаза (сдвига),  $\Delta$  — величина смаза, а  $w(x, y)$  — искомое распределение интенсивности на неискаженном изображении (той интенсивности, которая была бы на фотопленке в отсутствие смаза, т.е. при  $\Delta = 0$ ).

Соотношение (2.7) есть одномерное интегральное уравнение типа Вольтерра I рода относительно  $w(\xi, y)$  при каждом фиксированном значении  $y$ , играющем роль параметра, другими словами, (2.7) есть *совокупность (множество) одномерных интегральных уравнений*. Поэтому мы будем записывать (2.7) в виде (с учетом помехи на изображении) [96, 112]:

$$\frac{1}{\Delta} \int_x^{x+\Delta} w_y(\xi) d\xi = g_y(x) + \delta g, \quad (2.8)$$

где  $\delta g$  — помеха.

По терминологии работы [1], это уравнение является *неклассическим* уравнением Вольтерра I рода, так как оба предела интегрирования переменны. Интегральное уравнение (2.8) можно назвать также *нестандартным* (по терминологии работы [104]), так как оно не содержит ядро в явном виде (или ядро можно считать равным  $1/\Delta = \text{const}$ ).

Отметим, что в работах [4, 120] рассмотрены более сложные постановки задачи смазывания изображения: неравномерность перемещения фотоаппарата или объекта, а также наклонное расположение объекта.

Отметим еще, что значение  $\Delta$  часто априори неизвестно и его обычно определяют путем подбора на основе визуальной оценки получаемых для ряда значений  $\Delta$  решений  $w(\xi, y)$  [4, с. 164]. Кроме того, можно оценить величину, а также направление смаза по штрихам на снимке, особенно если хотя бы один из этих штрихов есть результат смазывания яркой точки на изображении, например, точки на текстовом изображении (см. рис. 2.6) или звезды на астрономическом снимке.

В результате, правильно выбрав направление оси  $x$  (вдоль смаза) и величину смаза  $\Delta$ , т.е. определив параметры смаза, можно, решив интегральное уравнение (2.8) (точнее, совокупность уравнений), восстановить в принципе неискаженный снимок (истинное изображение)  $w(x, y)$  по искаженному снимку-изображению  $g(x, y)$ . А искаженный снимок может быть, например, очень старым снимком, на котором заложена ценная, но трудно распознаваемая информация (нечеткая фотография знаменитой личности, исторического здания, важного текста и т.д.), и только математико-компьютерным путем можно будет извлечь эту информацию.

Рассмотрим один из методов решения уравнения (2.8).

**Метод дифференцирования** [102, с. 68], [109, 113]. Продифференцируем по  $x$  соотношение (2.8), получим:

$$\frac{1}{\Delta} [w_y(x + \Delta) - w_y(x)] = \frac{\partial \tilde{g}_y(x)}{\partial x},$$

где  $\tilde{g}_y(x) = g_y(x) + \delta g$ , откуда

$$w_y(x + \Delta) = w_y(x) + \Delta \frac{\partial \tilde{g}_y(x)}{\partial x}. \quad (2.9)$$

Выражение (2.9) есть *рекуррентное выражение*, согласно которому по предыдущему значению  $w_y(x)$  можно вычислить последующее значение  $w_y(x + \Delta)$ . Практически это означает следующее. Если при некотором  $x = x_0$  интенсивность  $w_y(x)$  априори известна (например, нулевая на краю снимка или в местах "провалов" в изображении на рис. 2.6), то с помощью (2.9) можно рекуррентно вычислить значения  $w_y(x_0 + \Delta)$ ,  $w_y(x_0 + 2\Delta)$ ,  $w_y(x_0 + 3\Delta)$ , ...

Запишем (2.9) иначе:

$$w_y(x) = w_y(x + \Delta) - \Delta \frac{\partial \tilde{g}_y(x)}{\partial x}$$

или, заменив  $x$  на  $x - \Delta$ ,

$$w_y(x - \Delta) = w_y(x) - \Delta \frac{\partial \tilde{g}_y(x - \Delta)}{\partial x}. \quad (2.10)$$

Тогда, используя (2.10), можно вычислить (двигаясь "справа налево", т.е. в сторону уменьшения  $x$ )  $w_y(x_0 - \Delta)$ ,  $w_y(x_0 - 2\Delta)$ ,  $w_y(x_0 - 3\Delta)$ , ... Операции (2.9) или (2.10) нужно выполнить при каждом значении  $y$ .

При этом производную  $\partial \tilde{g}_y(x) / \partial x$  следует вычислять путем *численного дифференцирования*, например, по формуле:

$$\frac{\partial \tilde{g}_y(x)}{\partial x} \approx \frac{\tilde{g}_y(x + \Delta x) - \tilde{g}_y(x - \Delta x)}{2\Delta x}, \quad (2.11)$$

где  $\Delta x$  — шаг численного дифференцирования по  $x$  (вообще говоря, отличный от  $\Delta$ ). Однако задача численного дифференцирования функции  $\tilde{g}_y(x)$ , измеренной с погрешностями, является *некорректной* (неустойчивой и т.д.) (о некорректных задачах см. дальше). Поэтому нужно использовать, например, метод регуляризации Тихонова для устойчивого дифференцирования зашумленных функций [50, с. 194], [117, с. 18–19, 158–159], а также сглаживать функцию  $\tilde{g}_y(x)$ , например, с помощью аппроксимирующих сплайнов [102, с. 223].

**Преобразование уравнения (2.8) к уравнению типа свертки.** Запишем ИУ (2.8) в виде:

$$\int_x^{x+\Delta} \frac{1}{\Delta} w_y(\xi) d\xi = \tilde{g}_y(x). \quad (2.12)$$

Преобразуем неклассическое и нестандартное ИУ (2.12) к ИУ Фредгольма I рода типа свертки [2, 4, 17, 96, 102, 109, 120, 168]:

$$\int_{-\infty}^{\infty} h(x - \xi) w_y(\xi) d\xi = \tilde{g}_y(x), \quad -\infty < x < \infty, \quad (2.13)$$

где

$$h(x - \xi) = \begin{cases} 1/\Delta, & x \leq \xi \leq x + \Delta \text{ или } -\Delta \leq x - \xi \leq 0, \\ 0, & \text{иначе,} \end{cases}$$

или

$$h(x) = \begin{cases} 1/\Delta, & -\Delta \leq x \leq 0, \\ 0, & \text{иначе.} \end{cases} \quad (2.14)$$

Здесь  $h$  — математически ядро ИУ, а физико-технически — *функция рассеяния точки* (ФРТ) или point spread function (PSF) [6, 17, 35, 36, 46, 137] — это то, во что превращается каждая точка при смазывании. В задаче смазывания функция  $h$  в большинстве случаев является разностной, или пространственно-инвариантной, что означает, что величина смаза  $\Delta$  не зависит от  $x$  (или  $\xi$ ).

Интегральное уравнение (2.13), наряду с уравнением (2.8), является основным в задаче реконструкции смазанного (и зашумленного) изображения.

**З а м е ч а н и е.** В задаче реконструкции дефокусированного (и зашумленного) изображения (см. далее п. 2.4) используется двухмерное интегральное уравнение Фредгольма I рода типа свертки [2, 26–28], [35, с. 80], [39, 102, 109, 112, 168]:

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x - \xi, y - \eta) w(\xi, \eta) d\xi d\eta = \tilde{g}(x, y) + \delta g, \quad (2.15)$$

причем уравнение такого типа используется в ряде работ ([2, 35, 39] и др.) для реконструкции не только дефокусированных, но и смазанных изображений, а также для реконструкции в случае более сложной ФРТ [39, 150]. При этом в случае равномерного и прямолинейного смаза вдоль одной координаты, например,  $x$  функция рассеяния точки может быть записана в виде [2]:

$$h(x - \xi, y - \eta) = h_1(x - \xi) \delta(y - \eta),$$

где  $h_1(x)$  выражается формулой (2.14), а  $\delta(y)$  есть  $\delta$ -функция. В этом случае двухмерное ИУ (2.15) примет вид одномерного ИУ типа (2.13) при каждом значении  $y$ , играющем роль параметра:

$$\int_{-\infty}^{\infty} h_1(x - \xi) w(\xi, y) d\xi = \tilde{g}(x, y) + \delta g.$$

Однако мы не будем стремиться к единому математическому аппарату, а именно, будем использовать двухмерное ИУ типа (2.15) лишь для реконструкции дефокусированных изображений, а задачу реконструкции смазанных изображений будем решать,

используя набор одномерных ИУ типа (2.13). Это упростит задачу реконструкции смазанных изображений, в частности, позволит избежать решения двухмерного ИУ с неразностной  $h$  (см. (2.61)) – весьма сложной задачи. Если же будет иметь место более сложный случай, чем смазывание или дефокусирование, то нужно использовать двухмерное ИУ [39, 150].

Для решения интегрального уравнения (2.13) обычно используются следующие *методы*: метод преобразования Фурье, или инверсной фильтрации [22, с. 256], [102, с. 182]; метод регуляризации Тихонова (прямой и итеративный варианты) [2, 4, 18, 22, 96, 102, 105, 109, 117, 120, 121, 144, 148, 167, 168]; метод оптимальной или параметрической фильтрации Винера [17, с. 106], [22, с. 315], [102, с. 210]; методы простой итерации (итеративной регуляризации) Ландвебера [160] и Фридмана [130], [22, с. 272], алгоритм Бочка итерационной реализации проекционных методов [13, 39], метод сопряженных градиентов (CG) [167] и др.

При численной реализации методов используются: дискретное (или быстрое) преобразование Фурье, квадратуры, итерации (последовательные приближения) и т.д.

**Об использовании "граничных условий" (BCs).** При этом часто используются, например, в работах [36, 144, 148, 150, 167] так называемые "граничные условия" (boundary conditions, BCs). Они вводятся для определения интенсивностей искомого изображения<sup>1</sup>  $w$  вне конечных границ искаженного изображения  $g$  (FOV — field of view [150, 167]), когда функция  $w$  не является финитной, т.е. носитель изображения в действительности шире FOV. При этом значения  $g$  вне FOV неизвестны и поэтому, как считают авторы работ [36, 144, 148, 150, 167] и др., не могут быть определены интенсивности  $w$  вне FOV путем решения уравнения типа (2.13) или (2.15). Между тем, интенсивности  $w$  вне FOV необходимы для решения как прямой задачи (моделирование  $g$ ), так и обратной задачи (реконструкция  $w$ ) — см. дальше. Поэтому были введены дополнительно следующие "граничные условия" (см. рис. 2.7):

- *zero, zero Dirichlet* (нулевые, нулевые по Дирихле) — полагается, что интенсивности  $w$  равны нулю вне границы  $B$  области  $D$  изображения  $g$  (FOV), т.е. искомая функция  $w$  полагается финитной (граничные условия типа дополнения нулями [143]);
- *symmetric, reflective, Neumann* (симметричные, рефлексивные) — интенсивности  $w$  вне границы  $B$  есть зеркальное отражение относительно  $B$  интенсивностей  $w$  внутри области  $D$  (данные "граничные условия" впервые введены в работе [166]);
- *replicate* (копирующие границу) — значения  $w$  вне  $B$  полагаются равными значениям  $w$  в соответствующих ближайших точках границы  $B$ ;
- *circular, periodic* (периодические) — значения  $w$  вне  $B$  периодически повторяются во всех направлениях (граничные условия типа периодического продолжения [143]);
- *anti-reflective* (анти-рефлексивные) — интенсивности  $w$  вне границы  $B$  полагаются сначала зеркальным отражением относительно  $B$ , а затем зеркальным отражением относительно нормали к  $B$  интенсивностей  $w$  внутри  $D$  (данные, наиболее эффективные, "граничные условия" впервые введены в работе [174]).

Различные "граничные условия" реализованы в MatLab в m-функции `imfilter.m`.

<sup>1</sup>Аналогичные *способы доопределения* изложены в книге Л.П. Ярославского [143, с. 103–107]: дополнение нулями, периодическое продолжение и четное продолжение.



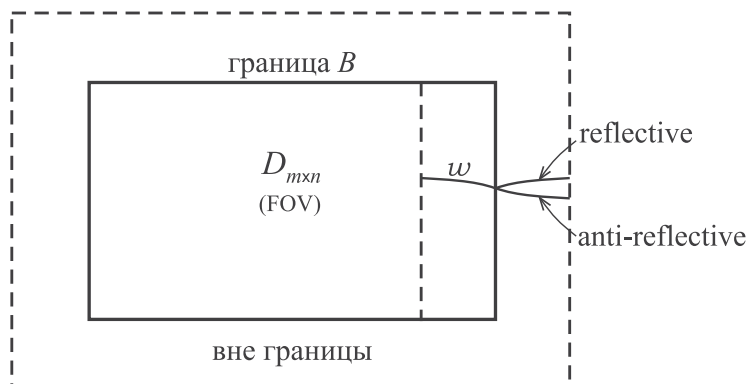


Рис. 2.7

Обычно под классическим термином *граничные условия* (или *краевые условия*) подразумеваются значения функции на некоторой границе  $B$  [16, 57, 71]. Однако авторы работ [36, 144, 148, 150, 167] в этот термин вложили другой смысл, а именно, доопределение значений функции  $w$  за границами  $B$  искаженного изображения  $g$ . Поэтому целесообразнее называть прием (способ) доопределения значений  $w$  за границами (вне границ)  $B$  изображения *внеграничными условиями* (off-boundary conditions, OBCs).

Внеграничные условия учитываются при решении прямой и обратной задач смазывания, а также дефокусирования (см. п. 2.5). В пакет Image Processing Toolbox (IPT) системы программирования MatLab 7 включен ряд программ в виде внешних м-функций<sup>2</sup> `deconvreg.m`, `deconvnr.m`, `deconvlucy.m` и `deconvblind.m` для реконструкции смазанных, а также дефокусированных изображений методами регуляризации, винеровской фильтрации, Люси–Ричардсона и "слепой" деконволюции с использованием OBCs [36, с. 183–195], [46]. Внеграничные условия можно считать различными вариантами *экстраполяции* значений  $w$  за границу  $B$ .

Непрерывным уравнениям (2.8) и (2.13) можно поставить в соответствие дискретное выражение вида:

$$g_y = Aw_y + \delta g, \quad (2.16)$$

где  $w_y$  — вектор-строка интенсивностей неискаженного изображения при некотором значении  $y$ ,  $g_y$  — вектор-строка интенсивностей смазанного изображения,  $\delta g$  — помеха,  $A$  — матрица, связанная с ФРТ и зависящая от типа внеграничных условий: в случае **zero** матрица  $A$  есть блочно-теплицева матрица с теплицевыми блоками (ВТТВ), в случае **symmetric (reflective)** — сумма ВТТВ-матрицы и блочно-ганкелевой матрицы с ганкелевыми блоками (ВННВ), в случае **circular (periodic)** — блочно-циркулянтная (циклическая) матрица с циркулянтными блоками (ВССВ) [167] и т.д.

Видим, что введение "граничных условий" (а более точно, внеграничных условий) является довольно искусственным приемом и к тому же их использование порождает

<sup>2</sup>Будем называть (ср. [44, с. 76], [45, с. 51]) функции типа `sin` *встроенными* (внутренними) функциями, функции типа `deconvnr.m` — *внешними* функциями, функции типа `smearing.m` (см. дальше) — *собственными* м-функциями, или м-файлами, а программы типа `text1.m` (см. дальше) — *головными программами*.

сложный математический аппарат. Вместо этого в работах [105,112] предложено более простое решение проблемы, названное *приемом (способом) усечения изображения*.

Заметим, что в некоторых случаях искомую функцию  $w$  все же можно считать финитной, например, когда рассматривается космический объект (см. рис. 2.4 и [167, Fig. 4]), текст ограниченных размеров (см. рис. 2.1) и т.д. Однако гораздо чаще искомая функция  $w$  не является таковой, как, например, в работах [26, рис. 2.4], [27, рис. 2.1], хотя авторы этих работ положили, что  $w$  финитна и, по-существу, использовали внеграницное условие **zero** (см. также [35, рис. 5.26a], [148, Fig. 7]).

Мы хотим показать, что в случае, когда  $w$  не является финитной, т.е.  $w$  отлична от нуля в более широкой области, чем область FOV, можно, тем не менее, адекватно решать как прямую, так и обратную задачу, не используя внеграницные условия, а используя так называемый *прием (способ) усечения изображения* [105,108,112].

**Прием усечения смазанного изображения.** Рассмотрим *прямую задачу* моделирования смазывания изображения. Пусть нам дано исходное (неискаженное) изображение размером  $m \times n$  пикселей, описываемое матрицей интенсивностей  $w_{m \times n}$ . Назовем область  $D_{m \times n}$  заданной *областью исходного изображения* (FOV — field of view), а границу  $B$  этой области — *границей исходного изображения* (см. рис. 2.7).

В редких случаях (когда, например, исходное изображение является изображением космического объекта) функцию  $w$  можно считать финитной, т.е. равной нулю вне области  $D$ . В этом случае можно без потери информации моделировать искаженное изображение по формуле типа (2.16), положив размеры матрицы  $g$  равными размерам матрицы  $w$  и используя при этом внеграницное условие **zero**. Аналогично, при решении *обратной задачи* (реконструкции изображения) можно без потери информации полагать размеры матриц  $w$  и  $g$  одинаковыми, т.е. считать, что  $w = 0$  вне области  $D$ .

Однако мы сконцентрируем внимание на более типичном случае, когда интенсивности  $w$  вне области  $D$ , вообще говоря, отличны от нуля, но информация о них отсутствует [144,167] (рис. 2.7).

В этом случае, как было уже сказано, часто используют (при моделировании искажения, а также при решении обратной задачи) внеграницные условия (OBCs). Вместо этого мы рассмотрим *прием (способ) усечения изображения* [105,108,112].

Рассмотрим простейший пример моделирования равномерного прямолинейного горизонтального смазывания изображения (прямая задача). Пусть нам дана матрица  $w_{m \times n}$  исходного (неискаженного) изображения и необходимо смоделировать матрицу  $g$  смазанного изображения при величине смаза  $\Delta$  и при нулевом угле смаза (горизонтальный смаз). Если формировать смазанное изображение такого же размера, как и исходное изображение, т.е. формировать  $g$  шириной  $n$ , то соотношение (2.16) более подробно следует записать в виде (при замене интеграла в (2.8) конечной суммой по пикселям, без помехи):

$$g_j(i) = \frac{1}{\Delta + 1} \sum_{k=i}^{i+\Delta} w_j(k), \quad i = 1, 2, \dots, n, \quad j = 1, 2, \dots, m, \quad (2.17)$$

где  $i$  ( $k$ ) — номер столбца,  $j$  — номер строки.

В (2.17) перед знаком суммы поставлен множитель  $1/(\Delta + 1)$  потому, что при  $\Delta = 0$  (в отсутствие смаза) выражение (2.17) должно переходить в  $g_j(i) = w_j(i)$ .<sup>3</sup> Для сравнения заметим, что в непрерывном (интегральном) выражении (2.8) перед знаком интеграла поставлен множитель  $1/\Delta$ , чтобы при  $\Delta \rightarrow 0$  выражение (2.8) переходило (без помехи) в  $w_y(x) = g_y(x)$  (подробнее см. в [102, с. 66]).

Из (2.17) видно, что при  $i = n - \Delta + 1, \dots, n$  требуется знание значений  $w_j(n - \Delta + 1), \dots, w_j(n + \Delta)$ , однако значения  $w_j(n + 1), \dots, w_j(n + \Delta)$  неизвестны. Чтобы преодолеть этот недостаток информации, в работах [36, 144, 148, 150, 167] и др., как уже было сказано, используются "граничные условия", которые мы называем внеграничными условиями: от простейшего **zero** до **anti-reflective**, в которых делаются различные предположения относительно недостающих значений  $w_j(n + 1), \dots, w_j(n + \Delta)$ .

В работах [105, 108, 112] предложено иное решение данного вопроса, а именно, вместо соотношений (2.17) предлагается использовать соотношения (см. рис. 2.8):

$$\hat{g}_j(i) = \frac{1}{\Delta + 1} \sum_{k=i}^{i+\Delta} w_j(k), \quad i = 1, 2, \dots, n - \Delta, \quad j = 1, 2, \dots, m. \quad (2.18)$$

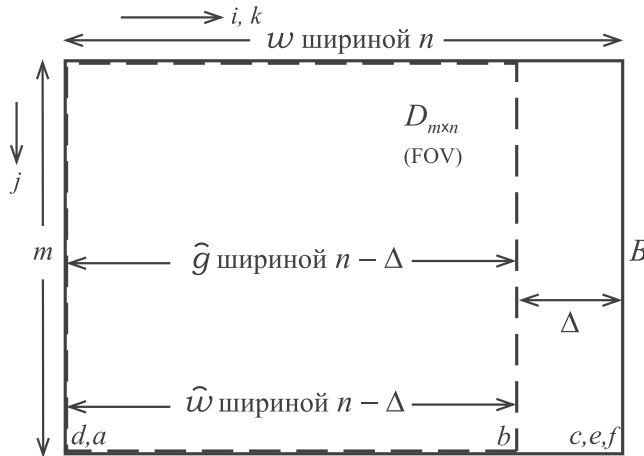


Рис. 2.8. Схема получения смазанного изображения с усечением (точки  $d, a, b, c, e, f$  приведены для сравнения с рис. 2.41, см. дальше)

Отличие выражения (2.18) от (2.17) состоит в том, что согласно (2.18) моделируется более узкое смазанное изображение. При этом требуются значения  $w_j(1), \dots, w_j(n)$ , которые известны, и не нужно прибегать к такому приему, как внеграничные условия. Правда,  $\hat{g}$  получается несколько уже, чем  $w$  (на  $\Delta$  пикселей). Но, во-первых, эффект заужения практически невелик (обычно  $n \approx 300 \div 600$ , а  $\Delta \sim 10$ ), а во-вторых, при

<sup>3</sup>Заметим, что интеграл в (2.8) заменен конечной суммой по пикселям (что соответствует физической природе явления смазывания), а не конечной суммой по некоторой квадратурной формуле, например, формуле трапеций — в последнем случае, кстати, нужно было бы учесть квадратурные коэффициенты под знаком суммы.

решении обратной задачи, например, методом регуляризации Тихонова эффективно решается недоопределенная система линейных алгебраических уравнений (СЛАУ), вытекающая из (2.18) (см. далее). Вариант моделирования смазанного изображения на основе (2.18) называется *приемом усечения смазанного изображения* [105, 112]. Данный прием (способ) реализован в собственной m-функции `smearing` (см. п. 2.4).

**Схема для понижения эффекта Гиббса.** Однако, как показало моделирование (см. п. 2.4, а также [108, 112]), схема с усечением (и не только эта схема) при решении обратной задачи часто порождает в восстановленном изображении искажения типа "звон" (ложные волны, эффект Гиббса) [17, с. 252], [36, с. 185], [108, 112, 165]. Это связано с тем, что функция  $\hat{g}_j(i)$  при  $i = 1$  и  $i = n - \Delta$  имеет разрывы непрерывности, так как  $\hat{g}_j(i) = 0$  при  $i \notin [1, n - \Delta]$  (типа функции Хэвисайда, или ступенчатой функции [102, с. 155], прямоугольного импульса, или стробирующей функции [15, с. 20]), а использование таких разрывных функций обычно ведет (в первую очередь, при использовании ПФ) к появлению флуктуаций типа функции sinc [17, с. 252], [102, с. 173–174], [165]. Чтобы подавить эти ложные флуктуации, можно использовать внешнюю m-функцию `edgetaper` системы MatLab (как рекомендуется в [36, с. 185]), размывающую края изображения  $\hat{g}$ . Однако, как показало моделирование (см. п. 2.4), эта функция недостаточно размывает края и более эффективный результат дает следующая схема с введением *размытых краев у изображения* (рис. 2.9) [108, 112]:

$$g_j(i) = \frac{1}{\Delta + 1} \sum_{k=i}^{i+\Delta} q_j(k), \quad i = 1, 2, \dots, n + \Delta, \quad j = 1, 2, \dots, m, \quad (2.19)$$

где

$$q_j(k) = \begin{cases} w_j(k - \Delta), & 1 \leq k - \Delta \leq n, \\ 0, & \text{иначе.} \end{cases}$$

Более сложная запись формулы (2.19) по сравнению с (2.18) объясняется ориентацией записи (2.19) на систему MatLab, в которой начальный индекс любого массива должен равняться 1. Можно запись (2.19) упростить, дополнив массив  $w$  слева и справа нулевыми матрицами размера  $m \times \Delta$ . В системе MatLab это можно выполнить, например, с помощью внешней m-функции `padarray`. Полученный массив обозначим через  $u$  (см. рис. 2.9). Тогда формулу (2.19) можно записать в более простом виде:

$$g_j(i) = \frac{1}{\Delta + 1} \sum_{k=i}^{i+\Delta} u_j(k), \quad i = 1, 2, \dots, n + \Delta, \quad j = 1, 2, \dots, m, \quad (2.20)$$

Такая запись будет особенно удобна при моделировании смазывания под углом с размыванием краев изображения (см. п. 2.4).

Схема с размытыми краями согласно (2.20) реализована в собственной m-функции `smearing.m` (см. п. 2.4).

Смазанное изображение  $g$  согласно (2.19) или (2.20) получается шириной  $n + \Delta$  с размытыми левым и правым краями (см. примеры дальше). Заметим, что введение

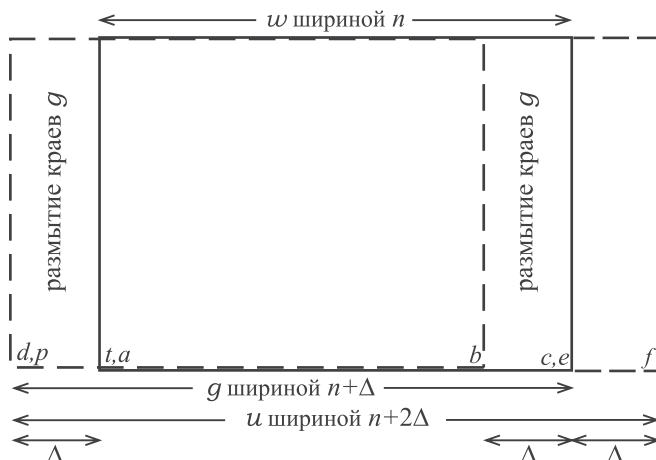


Рис. 2.9. Схема с введением размытых краев у изображения  $g$  (точки  $d, p, t, a, b, c, e, f$  приведены для сравнения с рис. 2.42, см. п. 2.4)

размытых краев не следует рассматривать как переход к финитности функции  $w$ . Введение размытых краев — это математический прием, направленный на умеренное сглаживание краев функции  $g$  и, как следствие, на понижение эффекта Гиббса на восстанавливаемом изображении  $w$  при решении обратной задачи (ср. [17, с. 252–253]).

Прием размытия краев у изображения напоминает сглаживание краев у передаточной функции  $H(\omega)$  прибора [17, с. 252], частотную фильтрацию сигналов с помощью рекурсивного фильтра, фильтра Гильберта и окна Кайзера [45, с. 216, 232, 236], а также сглаживание изображений с помощью низкочастотных фильтров: фильтра Баттерворта и гауссова фильтра [35, с. 265–269], [36, с. 143]. Однако в перечисленных фильтрах, особенно, в фильтре Баттерворта и гауссовом фильтре делается повышение гладкости краев передаточной функции фильтра  $H(\omega)$  или  $H(\omega_1, \omega_2)$  (см. далее (2.24) и (2.75)) в частотной области, что ведет к изменению фильтра  $h(x)$  или  $h(x, y)$  в пространственной области и к получению сглаженного изображения  $g_y(x) = h_y(x) * w_y(x)$  с возможным понижением эффекта Гиббса (см. [35, с. 261, рис. 4.12д]). А предлагаемый нами способ размытия отличается тем, что в нем осуществляется размытие краев непосредственно самого изображения  $g(x, y)$ , что ведет к большему размытию краев изображения (ср. [35, с. 261, рис. 4.12в] и рис. 2.12gp2) в прямой задаче и, как следствие, к большему подавлению эффекта Гиббса в обратной задаче.

Отметим, что размытие краев у изображения с целью понижения эффекта Гиббса можно выполнить в определенной степени также с помощью методов  $\sigma$ -множителей Ланцоша и Фейера [22, с. 259–260] или РКТ-метода Шеппа–Логана (см. (1.18)), но эти методы размывают не только края изображения, но и все изображение.

Можно выполнить эффективно размытие краев у изображений также с помощью опции 'full' внешней m-функции `imfilter.m`. Это дает такой же результат, как и с помощью собственной m-функции `smearing.m` (см. стр. 95) или `defocusing.m` (см. стр. 169), причем для изображений типа `gray` и `RGB`, класса `uint8`, `double` и др., смазанных и дефокусированных (типа рис. 2.12gp2, 2.43gir, 2.47gir и др., см. далее).

Тем не менее, мы практически не будем использовать `imfilter(...'full')`, а будем использовать `smearing(...'diffusion')` и `defocusing(...'diffusion'...)`, чтобы сохранить выбранный стиль обработки изображений.

В работе [35, с. 384–387] также изложен способ моделирования смазывания изображения (прямая задача), основанный на использовании двумерного преобразования Фурье и выполняющий следующие операции (см. (2.15)):

- 1)  $W(\omega_1, \omega_2)$  — ПФ от  $w(\xi, \eta)$ ,
- 2)  $H(\omega_1, \omega_2)$  — ПФ от  $h(x, y)$ ,
- 3)  $G(\omega_1, \omega_2) = H(\omega_1, \omega_2)W(\omega_1, \omega_2)$  — ПФ от  $g(x, y)$ ,
- 4)  $g(x, y)$  — обратное ПФ от  $G(\omega_1, \omega_2)$ .

В этом способе не используются "граничные условия" и размеры  $g$  равны размерам  $w$ . Однако он использует преобразование Фурье, к тому же двумерное. А операция ПФ — это операция, неадекватная физическому механизму смазывания, в котором присутствует только операция суммирования (сложения, накопления, усреднения) — см. (2.18) и (2.19). Поэтому в прямой задаче моделирования смазывания следует использовать преимущественно операцию суммирования (2.18), (2.19), но не преобразования Фурье (об обратной задаче см. далее).

**Обратная задача.** Рассмотрим задачу реконструкции смазанного (и зашумленного) изображения. Мы уже рассмотрели описание ее интегральным уравнением (2.8) и изложили метод дифференцирования решения этого уравнения. А сейчас остановимся на более распространенном варианте, когда задача реконструкции смазанного изображения описывается одномерным ИУ Фредгольма I рода типа свертки (2.13) относительно  $w_y(\xi)$  при каждом фиксированном значении  $y$ , играющем роль параметра. Рассмотрим несколько методов решения уравнения (2.13).

**Метод преобразования Фурье.** В *методе преобразования Фурье*, или *инверсной фильтрации* [17, с. 29, 71], [22, с. 256], [35, с. 388], [36, с. 183], [102, с. 69] решение уравнения (2.13), дающее распределение интенсивности на неискаженном изображении, записывается в виде обратного ПФ:

$$w_y(\xi) = \frac{1}{2\pi} \int_{-\infty}^{\infty} W_y(\omega) e^{-i\omega\xi} d\omega, \quad (2.21)$$

где ПФ (спектр) решения (в соответствии с теоремой о свертке [32]) равно

$$W_y(\omega) = \frac{G_y(\omega)}{H(\omega)}, \quad (2.22)$$

причем

$$G_y(\omega) = \int_{-\infty}^{\infty} \tilde{g}_y(x) e^{i\omega x} dx \quad (2.23)$$

— ПФ (спектр) правой части уравнения (2.13), или ПФ интенсивностей на смазанном

(и зашумленном) изображении  $\tilde{g}_y(x)$ , а

$$H(\omega) = \int_{-\infty}^{\infty} h(x) e^{i\omega x} dx \quad (2.24)$$

— ПФ (спектр) ядра уравнения, или ПФ функции рассеяния точки  $h(x)$ . Спектр  $H(\omega)$  обычно называют *оптической передаточной функцией* (ОПФ) или optical transfer function (OFT) [36, с. 155].

Преобразования Фурье  $G_y(\omega)$  и  $H(\omega)$  могут быть вычислены по некоторой стандартной программе вычисления ДПФ (или БПФ), например, по m-функции `fft.m` в системе MatLab. Впрочем,  $H(\omega)$  может быть вычислено и аналитически в виде НПФ с учетом (2.14):

$$H(\omega) = \int_{-\infty}^{\infty} h(x) e^{i\omega x} dx = \frac{1}{\Delta} \int_{-\Delta}^0 e^{i\omega x} dx = \frac{\sin(\omega\Delta)}{\omega\Delta} + \frac{\cos(\omega\Delta) - 1}{\omega\Delta} i. \quad (2.25)$$

Однако, учитывая, что ДПФ имеет ряд отличий от НПФ (периодичность и др. [102, с. 168]), целесообразнее вычислять  $H(\omega)$  через посредство ДПФ, как и  $G_y(\omega)$ .

Проанализируем поведение спектров  $G_y(\omega)$ ,  $H(\omega)$  и  $W_y(\omega)$ . При  $|\omega| \rightarrow \infty$  спектр  $G_y(\omega)$  правой части  $\tilde{g}_y(x)$  с учетом ее зашумленности стремится к некоторой константе (уровню "белого шума"), а спектр  $H(\omega) \rightarrow 0$  при  $|\omega| \rightarrow \infty$  (см. (2.25)) (ср. [17, с. 29–30]). Кроме того,  $H(\omega) = 0$  при  $\omega\Delta = 2\pi n$ ,  $n = \pm 1, \pm 2, \dots$ , т.е.  $H(\omega)$  имеет нули при конечных значениях  $\omega = 2\pi n/\Delta$ . Все это говорит о том, что решение методом инверсной фильтрации согласно (2.21)–(2.24) неустойчиво и сингулярно. Более того, задача решения уравнения (2.13) является *некорректной* (подробности см. дальше).

**Метод псевдоинверсной фильтрации.** Для устранения неустойчивости и сингулярностей в методе инверсной фильтрации предложен ряд простых модификаций этого метода, называемых иногда *псевдоинверсной фильтрацией* [36, с. 183]. Рассмотрим кратко наиболее распространенную модификацию — *решение в пространстве*  $S_{\pi/\Delta x}$  [22, с. 259], [125]. Пусть  $\tilde{g}_y(x)$  измерена лишь для дискретных значений  $x_i$  с шагом  $\Delta x = \text{const}$ . В этом случае, согласно теореме Котельникова [58], существует одна и только одна функция  $\tilde{g}_y(x)$ , принимающая в точках  $x_i$  измеренные значения  $\tilde{g}_y(x_i)$  и обладающая тем свойством, что ее спектр  $G_y(\omega) = 0$  при  $|\omega| > \omega_{\max}$ , где  $\omega_{\max} = \pi/\Delta x$  — частота Найквиста. Поэтому без потери информации, полученной при измерении, можно решение  $w_y(\xi)$  записать в более узком классе, чем (2.21):

$$w_y(\xi) = \frac{1}{2\pi} \int_{-\omega_{\max}}^{\omega_{\max}} W_y(\omega) e^{-i\omega\xi} d\omega. \quad (2.26)$$

Такое решение обладает некоторой устойчивостью, поскольку  $\omega_{\max}$  конечно. Теперь исследуем это решение на сингулярность. Чтобы  $H(\omega)$  обращалось в нуль, должно выполняться равенство:  $2\pi n/\Delta = (\pi/\Delta x) \cdot k/N$ , где  $N$  — число шагов дискретизации

по  $\omega$ , а  $k = 1, 2, \dots, N$  — номер узла дискретизации  $\omega_k$ , т.е.

$$2 \frac{N}{\Delta/\Delta x} n = k. \quad (2.27)$$

Чтобы левая часть равенства (2.27) принимала хотя бы одно из значений  $k = 1, 2, \dots, N$ , во-первых, отношение  $\Delta/\Delta x$  должно быть целым — это обычно выполняется, если  $x$  измеряется в пикселах, а во-вторых, должно быть целым и отношение  $N/(\Delta/\Delta x)$ . Например, при  $\Delta = 20$  пкс,  $\Delta x = 1$  пкс,  $N = 100$  равенство (2.27) примет вид:  $k = 10n$ , в результате при  $n = 1, 2, \dots, 10$ , т.е. при  $k = 10, 20, \dots, 100$ , другими словами, в каждом десятом узле дискретизации  $\omega_k$  спектр  $H(\omega_k)$  будет обращаться в нуль, а спектр решения  $W_y(\omega_k)$  (см. (2.22)) — в бесконечность. Значит, решение (2.26) (в виде ОДПФ) будет иметь сингулярности.

Если же отношение  $N/(\Delta/\Delta x)$  нецелое, например, при  $\Delta = 19$ ,  $\Delta x = 1$ ,  $N = 100$  или  $\Delta = 20$ ,  $\Delta x = 1$ ,  $N = 128$ , то  $H(\omega)$  не будет иметь нулей в узлах дискретизации  $\omega_k$ , но, тем не менее, значения  $H(\omega)$  могут быть близки к нулю.

Все говорит о том, что псевдоинверсная фильтрация (решение в пространстве  $S_{\pi/\Delta x}$ , а также использование низкочастотного фильтра Баттерворта, гауссова фильтра [35, с. 265–269, 389] и др.) недостаточно эффективна, но, тем не менее, она нередко используется для реконструкции изображений, в основном, в отсутствие шумов.

**Метод ПФ с регуляризацией Тихонова.** Гораздо более эффективным является решение ИУ (2.13) методом преобразования Фурье с *регуляризацией Тихонова*. Дадим его краткое изложение. Решение уравнения (2.13) методом ПФ с регуляризацией Тихонова имеет вид [2, 17, 22, 96, 102, 109, 112, 114, 115, 120, 168]:

$$w_{\alpha y}(\xi) = \frac{1}{2\pi} \int_{-\infty}^{\infty} W_{\alpha y}(\omega) e^{-i\omega\xi} d\omega, \quad (2.28)$$

где

$$W_{\alpha y}(\omega) = \frac{H(-\omega) G_y(\omega)}{|H(\omega)|^2 + \alpha \omega^{2p}} \quad (2.29)$$

— регуляризованный спектр решения,  $\alpha > 0$  — параметр регуляризации,  $p \geq 0$  — порядок регуляризации, обычно полагаемый равным  $p = 1, 2, 3, \dots$ , но может быть и нецелым (о выборе  $\alpha$  и  $p$  см. далее). Спектры  $G_y(\omega)$  и  $H(\omega)$  выражаются формулами (2.23) и (2.24) соответственно. Вычисления по формулам (2.28), (2.29) и (2.23) нужно выполнять при каждом значении  $y$  (при условии, что оси  $\xi$  и  $x$  направлены вдоль смаза, а ось  $y$  — перпендикулярно ему). Вычисление же  $H(\omega)$  согласно (2.24) нужно выполнять лишь один раз, так как  $H(\omega)$  не зависит от  $y$ .

Заметим, что при  $\alpha \rightarrow 0$  метод регуляризации Тихонова переходит в метод инверсной или псевдоинверсной фильтрации. Задача становится некорректной, а решение  $w$  — неустойчивым. В результате реконструированное изображение  $w$  получается даже худшего качества, чем искаженное изображение  $g$ . Однако при удачно выбранном  $\alpha$  изображение  $w_\alpha$  может получиться весьма близким к точному изображению  $\bar{w}$ .



В работах [35, с. 395], [36, с. 187] также изложен метод регуляризации Тихонова под названием фильтрация методом минимизации сглаживающего функционала со связью или сглаживающая фильтрация методом наименьших квадратов со связью.

Далее метод преобразования Фурье с регуляризацией Тихонова должен быть записан в дискретном виде, чтобы реализовать его на компьютере. При этом все прямые НПФ (2.23), (2.24) и (2.29), а также обратное НПФ (2.28) должны быть записаны в виде дискретных ПФ. Запись в виде ДПФ должна быть согласованной в том смысле, что носители (области, вообще говоря, ненулевых значений функций):  $\text{supp } h(x)$ ,  $\text{supp } w_y(\xi)$  и  $\text{supp } \tilde{g}_y(x)$  должны быть равной длины или даже совпадать [102, с. 201–202], [121, с. 39, 123]:

$$\text{supp } h(x) = \text{supp } w_y(\xi) = \text{supp } \tilde{g}_y(x).$$

В результате метод ПФ с регуляризацией Тихонова в дискретном виде был нами реализован в следующих двух вариантах [112].

**В а р и а н т 1.** В этом варианте используется усеченное смазанное изображение с матрицей интенсивностей  $\hat{g}$  размера  $m \times (n - \Delta)$  согласно (2.18). В этом случае матрица реконструированного изображения  $\hat{w}_\alpha$  при дискретной реализации формулы (2.28), а также формул (2.23), (2.24) и (2.29) будет иметь такой же размер, как и матрица  $\hat{g}$ , а именно,  $m \times (n - \Delta)$ .

**В а р и а н т 2.** В этом варианте используется усеченное смазанное изображение с размытыми краями с матрицей интенсивностей  $g$  размера  $m \times (n + \Delta)$  согласно (2.19). В этом случае матрица реконструированного изображения  $w_\alpha$  будет иметь размер  $m \times (n + \Delta)$ . Однако левый край матрицы  $w_\alpha$  шириной  $\Delta$  пикселей играет лишь вспомогательную роль, связанную с подавлением эффекта Гиббса, а к восстанавливаемому изображению не имеет отношения. Поэтому после расчета матрицы  $w_\alpha$  размера  $m \times (n + \Delta)$  делается удаление ее левого края на  $\Delta$  пикселей, в результате чего размер матрицы  $w_\alpha$  станет равным  $m \times n$ .

Варианты 1 и 2 реализованы в собственной m-функции `desmearingf.m` (см. п. 2.4) — реконструкция смазанного изображения методом ПФ с регуляризацией Тихонова с использованием приема "усечение" или "размытие краев" изображения.

**Метод конечных сумм с регуляризацией Тихонова.** Методы решения обратных задач, использующие преобразование Фурье, изложенные выше (а также метод оптимальной и параметрической фильтрации Винера — см. далее), обладают тем принципиальным недостатком, что они не вполне адекватны физической природе смазывания изображения. А именно, в процедуре моделирования смазывания изображения (прямой задаче) используется лишь операция суммирования (см. суммы в выражениях (2.18) и (2.19)). Такая же операция суммирования (интегрирования) используется самой природой, когда реальной системой наблюдений (фотоаппаратом, телескопом и т.д.) получается реальное смазанное изображение (см. (2.6)). Если же при решении обратной задачи использовать, помимо суммирования, другие математические операции: экспоненты, косинусы, синусы, умножение, деление, как это имеет место в методах, использующих ПФ (см. (2.21)–(2.29)), то решение типа (2.26), (2.28)

и т.п. будет обладать пониженной точностью по сравнению с методами, в которых в обратной задаче используются, как и в прямой задаче, лишь операции суммирования. Другими словами, обратная и прямая задачи должны быть взаимно согласованными математически.

Таким методом решения обратной задачи является метод конечных сумм<sup>4</sup> (а также метод итераций и др.).

Запишем уравнение типа свертки (2.13) в виде уравнения общего типа [17, с. 14], [96, 112]

$$Aw_y \equiv \int_a^b h(x, \xi) w_y(\xi) d\xi = \tilde{g}_y(x), \quad c \leq x \leq d, \quad (2.30)$$

где

$$h(x, \xi) = \begin{cases} 1/\Delta, & x \leq \xi \leq x + \Delta, \\ 0, & \text{иначе.} \end{cases} \quad (2.31)$$

Здесь  $A$  — интегральный оператор.

**В а р и а н т 3.** Учтем, что все величины  $x, \xi, y, a, b, c, d, \Delta$  выражаются в пикселах, т.е. принимают целые значения, а применительно к системе MatLab — натуральные значения  $1, 2, 3, \dots$

Интеграл в (2.30) заменим конечной суммой по пикселям, а координатам припишем натуральные значения. Рассмотрим сначала схему с усечением (2.18). Имеем:

$$Aw_j \equiv \sum_{k=1}^n \hat{h}_{ik} w_{j,k} = \hat{g}_{j,i}, \quad i = 1, \dots, n - \Delta, \quad j = 1, \dots, m, \quad (2.32)$$

где

$$\hat{h}_{ik} = \begin{cases} 1/(\Delta + 1), & i \leq k \leq i + \Delta, \\ 0, & \text{иначе.} \end{cases} \quad (2.33)$$

**В а р и а н т 4.** Теперь обратимся к схеме с размытыми краями (2.19). Получим:

$$Aw_j \equiv \sum_{k=1}^n h_{ik} w_{j,k} = g_{j,i}, \quad i = 1, \dots, n + \Delta, \quad j = 1, \dots, m, \quad (2.34)$$

$$h_{ik} = \begin{cases} 1/(\Delta + 1), & i - \Delta \leq k \leq i, \\ 0, & \text{иначе.} \end{cases} \quad (2.35)$$

**В а р и а н т 5.** Рассмотрим для полноты также вариант двойного усечения, когда в схеме на рис. 2.8 усекается как  $g$ , так и  $w$  до размера  $m \times (n - \Delta)$ :

$$A\hat{w}_j \equiv \sum_{k=1}^{n-\Delta} \hat{h}_{ik} \hat{w}_{j,k} = \hat{g}_{j,i}, \quad i = 1, \dots, n - \Delta, \quad j = 1, \dots, m, \quad (2.36)$$

---

<sup>4</sup>Мы будем называть этот метод также методом квадратур, хотя название "метод конечных сумм" (подразумевающий суммирование по пикселям матрицы изображения) больше соответствует физико-технической сути задачи.

где  $\widehat{h}_{ik}$  выражается формулой (2.33).

Проанализируем варианты 3–5 (формулы (2.32)–(2.36)). В соотношениях (2.32), (2.34) и (2.36)  $A$  — это ленточная матрица, состоящая из элементов  $\widehat{h}_{ik}$  или  $h_{ik}$  и независящая от номера строки  $j$ . При этом  $A$  — это матрица размера  $(n - \Delta) \times n$  в (2.32),  $(n + \Delta) \times n$  в (2.34) и  $(n - \Delta) \times (n - \Delta)$  в (2.36). При каждом фиксированном номере строки  $j$  мы получаем СЛАУ относительно искомого вектора  $w$  или  $\widehat{w}$  (напоминаем, что для каждого значения  $j$  матрица СЛАУ одна и та же).

При этом СЛАУ (2.32) является недоопределенной (измеренный вектор  $\widehat{g}$  имеет меньшую ширину, чем искомый вектор  $w$ , см. рис. 2.8, следовательно, число уравнений  $n - \Delta$  меньше числа неизвестных  $n$ ). СЛАУ (2.34) является переопределенной (вектор  $g$  имеет ббольшую ширину, чем вектор  $w$ , см. рис. 2.9, следовательно число уравнений  $n + \Delta$  больше числа неизвестных  $n$ ). Наконец, СЛАУ (2.36) является определенной (вектор  $\widehat{g}$  имеет такую же ширину, как и искомый вектор  $\widehat{w}$ , см. рис. 2.8, следовательно, число уравнений  $n - \Delta$  равно числу неизвестных  $n - \Delta$ ). Чтобы реконструировать все изображение, нужно при каждом  $j$  (т.е.  $m$  раз) решить подобную СЛАУ. Другими словами, в нашем подходе (подразумеваемом, что ось  $x$  или  $i$  направлена вдоль смаза) реконструкция изображения выполняется построчно.

Более кратко запишем СЛАУ (2.32), (2.34), (2.36) в виде:

$$Aw = g. \quad (2.37)$$

В работе [112] решение интегрального уравнения (2.30) путем приведения его к СЛАУ согласно (2.32)–(2.37) названо методом квадратур. Однако более точно его следует называть *методом конечных сумм*, поскольку имеет место суммирование, причем не по математическим дискретным квадратурным узлам, а по реальным техническим пикселям.

Важной особенностью является то, что задача решения интегрального уравнения (2.30) (как и уравнения (2.13)), а также СЛАУ (2.32), (2.34) и (2.36) является *некорректной*, а именно, нарушается хотя бы одно из условий *корректности* по Адамару [50, с. 9], [66, с. 5], [115, с. 11]:

- 1) решение существует,
- 2) решение единственно,
- 3) решение устойчиво.

С этой точки зрения, недоопределенная СЛАУ (2.32) имеет множество решений (нарушается 2-е условие корректности), а переопределенная СЛАУ (2.34) не имеет решения (нарушается 1-е условие корректности). И, самое главное, решения уравнения (2.30), а также СЛАУ (2.32), (2.34), (2.36), (2.37) сильно неустойчивы (см. далее рис. 2.27, а также [102, с. 65, рис. 2.36]).

Чтобы преодолеть эти сложности, мы воспользуемся методом регуляризации Тихонова (подробности см. на с. 207).

Решение СЛАУ (2.37) *методом регуляризации Тихонова* имеет вид [22, 93, 96, 102, 112, 114, 115, 148, 150, 168]:

$$w_\alpha = (\alpha E + A^T A)^{-1} A^T g, \quad (2.38)$$

где  $E$  — единичная матрица,  $A^T$  — транспонированная матрица.

**З а м е ч а н и е.** Регуляризацию вида (2.38) принято называть *классической тихоновской регуляризацией* [20]. Она соответствует минимизации сглаживающего функционала [21, с. 20]

$$\|Aw - g\|_{L_2}^2 + \alpha\Omega[w] = \min_w$$

со стабилизатором  $\Omega[w] = \|w\|_{L_2}^2$ . К классической тихоновской регуляризации относят также регуляризацию со стабилизатором  $\Omega[w] = \|w\|_{W_2^1}^2$  [21, с. 21], [22, с. 240] или  $\Omega[w] = \|w\|_{W_2^2}^2$  (стабилизатор в пространстве Соболева, задаваемый, в частности, в виде лапласиана [35, с. 397], [36, с. 187]).

Однако интенсивность  $w$  на изображении является, как правило, негладкой (недифференцируемой) функцией. Поэтому классическая регуляризация со стабилизатором вида  $\Omega[w] = \|w\|_{W_2^2}^2$  не вполне подходит для реконструкции искаженных изображений (такая регуляризация будет излишне заглаживать изображение и будет теряться его тонкая структура) [20], [21, с. 21]. Даже регуляризация со стабилизатором  $\Omega[w] = \|w\|_{L_2}^2$  (назовем ее *классической  $L_2$ -тихоновской регуляризацией*) ведет, согласно [180], к большему заглаживанию изображения, а значит, к меньшему разрешению и резкости, чем, например,  $L_1$ -регуляризация такая, как bilateral total variation method (BTV method, метод двусторонней полной вариации) [152] или total variation method (TV method) [171].

Еще раньше, начиная с 70-х годов прошлого столетия, был опубликован ряд работ (см. ссылки в [20, 21]), в которых были предложены стабилизаторы вида

$$\Omega[w] = \sigma\|w\|_{L_p}^p + \varkappa\|w\|_{V_a^b}, \quad p \geq 1, \quad \sigma^2 + \varkappa^2 > 0,$$

где использовалась *вариация функции* (в 1-мерном случае)

$$\|w\|_{V_a^b} \equiv V_a^b(w) = |w(a)| + \sup_{T_n} \left\{ \sum_{i=1}^n |w(x_i) - w(x_{i-1})| \right\},$$

причем  $T_n$  — дискретная сетка узлов, или (вариация Витали в 2-мерном случае)

$$\Omega[w] = V(w) = \sup_{T_n, T_m} \left\{ \sum_{i=1}^{n-1} \sum_{j=1}^{m-1} |w(x_{i+1}, y_{j+1}) + w(x_i, y_j) - w(x_i, y_{j+1}) - w(x_{i+1}, y_j)| \right\}.$$

Однако в данной книге мы будем пользоваться классической  $L_2$ -тихоновской регуляризацией, добавляя к ней для повышения эффективности способы "усечения изображения" (см. стр. 82), "размытия краев изображения" (см. стр. 84) и "двойного поворота изображения" (см. дальше стр. 139), а также для фильтрации шумов адаптивную фильтрацию Винера, медианную и ранговую фильтрацию (см. дальше стр. 117). Кроме того, не будем использовать мультикадровую обработку, как в работах [152, 171, 180] и др. Другими словами, постараемся взять из классической  $L_2$ -тихоновской регуляризации максимум возможного (при условии добавления вспомогательного материала, а именно, новых способов "усечения–размытия–поворота изображения" и известных методов фильтрации шумов).

Итак, обратимся к методу  $L_2$ -регуляризации Тихонова. Отличительной особенностью этого метода является то, что он дает возможность решить недоопределенную, переопределенную и определенную СЛАУ, давая приближение к точному решению, в качестве которого выступает *нормальное псевдорешение*; оно существует и является единственным, т.е. выполняются 1-е и 2-е условия корректности. Кроме того, при удачно выбранном параметре регуляризации  $\alpha$  решение  $w_\alpha$  является *устойчивым*, т.е. выполняется 3-е условие корректности.

Варианты 3, 4 и 5 реализованы в собственной m-функции `desmearingq.m` (см. п. 2.4) — реконструкция смазанного изображения методом конечных сумм (квадратур) с регуляризацией Тихонова.

**Метод фильтрации Винера.** Для сравнения изложим (кратко) также метод фильтрации Винера решения ИУ типа свертки (2.13) [17, с. 105–113], [35, с. 391], [36, с. 183], [102, с. 211]. Согласно нему, решение уравнения (2.13) (неискаженное изображение) равно (ср. (2.28), (2.29))

$$w_y(\xi) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{H(-\omega) G_y(\omega)}{|H(\omega)|^2 + S_n(\omega)/S_{w_y}(\omega)} e^{-i\omega\xi} d\omega, \quad (2.39)$$

где  $S_n(\omega)$  и  $S_{w_y}(\omega)$  — спектральные мощности шума  $n = \delta g$  и искомого решения (усреднение по ансамблю реализаций мощностных, или энергетических спектров шума и решения). Метод, использующий соотношение (2.39), называется методом *оптимальной линейной фильтрации Винера*. Однако в таком виде данный метод практически нереализуем, так как  $S_n(\omega)$  и, особенно,  $S_{w_y}(\omega)$  обычно неизвестны. Если пойти на упрощение и использовать лишь одну реализацию энергетических спектров шума и решения  $|N(\omega)|^2$  и  $|W_y(\omega)|^2$ , то  $S_n(\omega)/S_{w_y}(\omega) = |N(\omega)|^2/|W_y(\omega)|^2$  и такой вариант метода называется *винеровской фильтрацией*. При этом отметим, что если шум равен нулю, то  $|N(\omega)|^2 = 0$  и винеровская фильтрация переходит в инверсную фильтрацию (см. (2.21)–(2.24), если числитель и знаменатель (2.22) умножить на  $H(-\omega)$ ).

Однако если спектр шума  $N(\omega)$  можно иногда определить, то спектр решения (неискаженного изображения)  $W_y(\omega)$  бывает известен лишь в модельных примерах, а при обработке реального искаженного изображения он обычно неизвестен. В этом случае выражение часто записывают в виде [35, с. 392], [36, с. 184]:

$$w_{Ky}(\xi) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{H(-\omega) G_y(\omega)}{|H(\omega)|^2 + K} e^{-i\omega\xi} d\omega, \quad (2.40)$$

где  $K \geq 0$  — некоторая константа (параметр). Такое решение называется *параметрической винеровской фильтрацией*.

Винеровская фильтрация, в частности, параметрическая винеровская фильтрация реализована в системе программирования MatLab 7 в виде внешней m-функции `deconvwnr`. Далее мы неоднократно будем обращаться к функции `deconvwnr`. При этом следует иметь в виду, что в различных подверсиях версии MatLab 7, например, в MatLab 7.0.1 и MatLab 7.10.0 реализованы несколько различные варианты функции

`deconvnr`, которые при малых значениях параметра  $K$  могут давать заметно различные решения  $w_{Ky}(\xi)$ . К этому вопросу мы еще будем возвращаться.

### Контрольные задания и вопросы

1. Что такое прямая задача в реконструкции смазанных изображений? Сформулируйте кратко постановку прямой задачи на примере фотоаппарата.
2. Используя формулы (2.5) и (2.4), покажите, что  $f_1 \geq f$  и  $f_2 \geq f$ .
3. Сформулируйте обратную задачу реконструкции смазанных изображений. Чем она отличается от прямой задачи?
4. Классифицируйте уравнение (2.7) или (2.8) (одномерное или двухмерное и т.д.).
5. Дайте подробный вывод (2.9) путем дифференцирования (2.8). В чем состоят особенности метода дифференцирования на основе (2.9)–(2.11)?
6. Обоснуйте переход от (2.12) к (2.13)–(2.14). В чем состоят преимущества уравнения (2.13) по сравнению с (2.12)?
7. Как Вы понимаете физически, технически и математически функцию рассеяния точки (ФРТ)? Что означает пространственная инвариантность ФРТ?
8. В чем состоит различие и сходство уравнений (2.13) и (2.15)?
9. Чем вызвана необходимость использования так называемых "граничных условий"? Перечислите их типы. Какое более удачное название можно предложить?
10. Какие внешние m-функции включены в MatLab 7 для решения прямой и обратной задач?
11. Изложите идею приема "усечение смазанного изображения". Требуется ли этот прием использования "граничных условий"? Чем отличается (2.18) от (2.17)?
12. Объясните, чем обусловлено появление на реконструируемых изображениях искажений типа "звоны" (ложные волны, эффект Гиббса)? Почему размытие краев искаженного изображения ведет к значительному понижению эффекта Гиббса? Прокомментируйте (2.19) и (2.20) и рис. 2.9.
13. Перечислите основные методы решения ИУ (2.13) и дайте им краткую характеристику. В чем состоит сходство и различие метода регуляризации Тихонова (см. (2.28), (2.29)) и метода фильтрации Винера (см. (2.39), (2.40))? Чем отличаются варианты 1 и 2 метода ПФ с регуляризацией?
14. Укажите основные отличия метода конечных сумм (с регуляризацией Тихонова) от метода ПФ (также с регуляризацией Тихонова). Охарактеризуйте варианты 3–5 метода конечных сумм (формулы (2.32)–(2.36)).

## 2.4. Реконструкция смазанных и зашумленных изображений (численные иллюстрации)

**Программы в системе MatLab.** В рамках системы программирования MatLab 7 был разработан ряд собственных m-функций и головных программ [7, 107]. Их условно можно разделить на две группы.

**Первая группа** m-функций и головных программ предназначена для моделирования *прямой задачи* смазывания изображений согласно выражениям (2.18),

(2.20), а также их зашумления. В первой группе основными являются собственные м-функции `smearing.m` и `normnoise.m`. Дадим их краткое описание (полные тексты см. на прилагаемом компакт-диске CD-R).

```
function [g,gir] = smearing(w,Delta,theta,type)
%Моделирование смазанного под углом gray- или RGB-изображения
%с усечением изображения (type='truncation') или с размытием
%краев (type='diffusion') - прямая задача.
%Входные параметры (аргументы):
%w - матрица изображения типа gray или RGB размера mхn класса double,
%Delta - величина смаза в пкс (double),
%theta - угол смаза в град против час стрелки
% (если theta<0, то по час стрелке),
%type - если type='truncation', то моделируется смазанное
% изображение с усечением (лишь при theta=0), а если
% type='diffusion', то с размытием краев (при произвольном theta).
%Выходные параметры (аргументы):
%g - смазанное горизонтально gray- или RGB-изображение класса
% double размера mх(n-Delta), если type='truncation', или
% повернутое на угол theta и смазанное gray- или RGB-изображение
% размера (m|cos(t)|+n|sin(t)|) x (n|cos(t)|+m|sin(t)|+Delta),
% если type='diffusion', где t=theta*pi/180,
% при любом theta смазывание на g будет идти горизонтально,
%gir - смазанное и повернутое обратно gray- или RGB-изображение
% размера (m+Delta|sin(t)|) x (n+Delta|cos(t)|)
% (в частном случае, когда theta=0, gir совпадает с g),
% изображение gir используется лишь для демонстрационных целей,
%g и gir - такого же типа (gray или RGB), как и w.
%Возможные варианты обращений к smearing:
%g = smearing(w,Delta,0,'truncation'); %theta=0
%или
%g = smearing(w,Delta,0,'diffusion'); %theta=0
%или
%g = smearing(w,Delta,theta,'diffusion'); %theta произволен
%или
%[g,gir] = smearing(w,Delta,theta,'diffusion'); %theta произволен

function gn = normnoise(g,per)
%Добавление к изображению нормального (гауссова) шума
%с нулевым средним значением.
%Вход:
%g - исходное изображение типа gray или RGB
% класса uint8, uint16 или double,
```

```
%per - уровень шума в процентах.
%Выход:
%gn - зашумленное с помощью функции randn
%      изображение (такого же типа и класса, что и g).
```

**З а м е ч а н и е.** Зашумление изображения гауссовым шумом можно выполнить также с помощью внешней *m*-функции `imnoise` в форме: `imnoise(g, 'gaussian', m, v)` [36, с. 156], [45, с. 548]. Однако [36, с. 156] обращение к `imnoise` связано с некоторыми неудобствами, а именно, с необходимостью нормировки среднего значения *m* и дисперсии *v*. В разработанной нами *m*-функции `normnoise` таких неудобств нет. Что же касается зашумления изображения импульсным шумом (типа "соль и перец"), то *m*-функция `imnoise` в форме: `imnoise(g, 'salt & pepper', d)` [36, с. 156], [45, с. 549] достаточно удобна, поэтому для зашумления изображения импульсным шумом мы использовали *m*-функцию `imnoise(g, 'salt & pepper', d)`.

В первую группу входит также собственная *m*-функция `normL2.m`:

```
function n2 = normL2(g)
%Норма в пространстве L2 (используется лишь в normnoise).
%g - входной массив (вектор, 2d- или 3d-матрица)
%      класса uint8 или double,
%n2=normL2 - выход (скаляр класса double).
```

К первой группе относятся также первые половины головных программ `text1.m`, `text2.m`, `text3.m`, `text3err.m`, `text3Werr.m`, `text4.m`, `text4err.m`, `text4Werr.m`, `text5.m`, `text6.m`, `text7.m` и `text8.m`.

**В т о р а я г р у п п а** *m*-функций и головных программ предназначена для решения *обратной задачи* — реконструкции смазанных и зашумленных изображений согласно соотношениям (2.13), (2.14), (2.23), (2.24), (2.28), (2.29), записанным в дискретном виде (варианты 1, 2) и (2.30)–(2.38) (варианты 3–5), а также (2.40) (для сравнения). Во вторую группу входят собственные *m*-функции `desmearingf.m` и `desmearingq.m`. Приведем их краткое описание.

```
function w_alpha = desmearingf(g,Delta,theta,variant,alpha,p,occasion,m,n)
%Реконструкция смазанного под углом модельного или реального изображения
%типа gray или RGB методом преобразования Фурье (ПФ) и рег-ции Тихонова.
%Если изображение модельное, то решается вариант 1 (с усечением) или
%вариант 2 (с размытием), причем вариант 1 решается лишь при theta=0,
%а вариант 2 при произвольном theta. Если же изображение реальное, то
%решается лишь вариант 1 (с усечением при произвольном theta).
%Входные параметры (аргументы):
%g - смазанное изображение типа gray или RGB класса double или uint8
%      размера mgxng, причем mg=m, ng=n-Delta в варианте 1 и
%      mg=m|cos(t)|+n|sin(t)|, ng=n|cos(t)|+m|sin(t)|+Delta в варианте 2,
%      где t=mod(theta,180), а mxn - размер истинного изображения w,
%Delta - величина смаза в пкс (double),
```



```

%theta - угол смаза в град против час стрелки
% (если theta<0, то по час стрелке),
%variant - строка символов (вариант решения):
% если variant='var1', то решается вариант 1 - усечение изображения
% (лишь при theta=0, если изображение модельное, и при произвольном
% theta, если изображение реальное), а если variant='var2',
% то решается вариант 2 - размытие (при произвольном theta,
% изображение только модельное);
% замечание: можно писать 'var1' или 'var 1' или 'variant1' или
% 'variant 1' или '1' и т.п. (главное, чтобы в конце была цифра 1),
% аналогично 'var2' или 'var 2' или '2' и т.п.,
%alpha>0 - параметр регуляризации (double),
%p>0 - порядок регуляризации (double)
% (обычно p=1,2,3,..., но может быть и нецелым),
%occasion - строка символов (случай получения изображения g):
% если occasion='model', то полагается, что изображение было
% получено в результате моделирования смаза по m-файлу smearing.m,
% причем оно было повернуто на угол theta (с помощью imrotate.m)
% и добавлено нулями с помощью padarray.m (в случае размытия);
% если occasion='real', то вариант может быть 1 или 2,
% а если occasion='real', то изображение было получено
% в результате реально сделанного снимка со смазом, при этом
% изображение будет специально повернуто в данной функции с помощью
% imrotate.m; если occasion='real', то вариант может быть только 1;
% можно писать: 'model' или 'm' или 'real' или 'r' и т.д.,
% главное, чтобы первый символ был m или r,
%m и n - количество строк и столбцов у истинного изображения w
% (m и n нужны для удаления черной рамки вокруг w_alpha в
% варианте 2 при theta~=0; если обрабатывается реальное
% изображение, когда w неизвестно, то m и n нужно подобрать).
%Выходной параметр (аргумент):
%w_alpha - реконструированное изображение типа gray или RGB
% класса double размера mnxng, если variant='var1',
% и размера mxn, если variant='var2'.

function w_alpha = desmearingq(g,Delta,theta,variant,alpha,occasion,m,n)
%Реконструкция смазанного под углом модельного или реального изображения
%типа gray или RGB методом квадратур и регуляризации Тихонова.
%Если изображение модельное, то решается недоопределенная или
%определенная СЛАУ - варианты 3 или 5 (с усечением, лишь при theta=0)
%или переопределенная СЛАУ - вариант 4 (с размытием при произвольном
%theta). Если же изображение реальное, то решается лишь определенная
%СЛАУ - вариант 5 (с усечением при произвольном theta).

```

```

%Входные параметры (аргументы):
%g - смазанное изображение типа gray или RGB класса double или uint8
%   размера mgxng, причем  $mg=m|\cos(t)|+n|\sin(t)|$ ,
%    $ng=n|\cos(t)|+m|\sin(t)|+\Delta$  в варианте 4 и
%    $mg=m$ ,  $ng=n-\Delta$  в варианте 3 или 5, где
%    $t=\text{mod}(\text{theta},180)$ , а  $m \times n$  - размер истинного изображения  $w$ ,
%Delta - величина смаза в пкс (double),
%theta - угол смаза в град против час стрелки
%   (если  $\text{theta}<0$ , то по час стрелке),
%variant - строка символов (вариант решения):
%   если  $\text{variant}='var3'$ , то решается недоопределенная СЛАУ -
%   вариант 3 (лишь при  $\text{theta}=0$ , изображение модельное),
%   если  $\text{variant}='var4'$ , то решается переопределенная СЛАУ -
%   вариант 4 (при произвольном  $\text{theta}$ , изображение модельное),
%   если  $\text{variant}='var5'$ , то решается определенная СЛАУ -
%   вариант 5 (лишь при  $\text{theta}=0$ , если изображение модельное, и
%   при произвольном  $\text{theta}$ , если изображение реальное);
%   замечание: можно писать 'var3' или 'var 3' или 'variant3' или
%   'variant 3' или '3' и т.п. (главное, чтобы в конце была цифра 3),
%   аналогично 'var4', '4', 'var5', '5' и т.п.,
%alpha>0 - параметр регуляризации (double),
%occasion - строка символов (случай получения изображения g):
%   если  $\text{occasion}='model'$ , то полагается, что изображение было
%   получено в результате моделирования смаза по  $m$ -файлу smearing.m,
%   причем оно было повернуто на угол  $\text{theta}$  (с помощью imrotate.m)
%   и добавлено нулями с помощью padarray.m (в случае размытия);
%   если  $\text{occasion}='model'$ , то вариант может быть 3, 4 или 5,
%   а если  $\text{occasion}='real'$ , то изображение было получено
%   в результате реально сделанного снимка со смазом, при этом
%   изображение будет специально повернуто в данной функции с помощью
%   imrotate.m; если  $\text{occasion}='real'$ , то вариант может быть только 5;
%   можно писать: 'model' или 'm' или 'real' или 'r' и т.д.,
%   главное, чтобы первый символ был  $m$  или  $r$ ,
% $m$  и  $n$  - количество строк и столбцов у истинного изображения  $w$ 
%   ( $m$  и  $n$  нужны для удаления черной рамки вокруг  $w\_alpha$  в
%   варианте 4 при  $\text{theta} \sim 0$ ; если обрабатывается реальное
%   изображение, когда  $w$  неизвестно, то  $m$  и  $n$  нужно подобрать).
%Выходной параметр (аргумент):
%w_alpha - реконструированное изображение типа gray или RGB
%   класса double размера  $m \times n$ , если  $\text{variant}='var3'$  или  $\text{variant}='var4'$ ,
%   и размера  $m \times (n-\Delta)$ , если  $\text{variant}='var5'$ .

```

Ко второй группе относятся также вторые половины головных программ `text1.m`, `text2.m`, `text3.m`, `text3err.m`, `text3Werr.m`, `text4.m`, `text4err.m`, `text4Werr.m`,

text5.m, text6.m, text7.m и text8.m.

При решении обратной задачи (с помощью второй группы  $m$ -функций и головных программ) параметр регуляризации  $\alpha$  в методе регуляризации Тихонова (см. (2.28), (2.29), (2.38)) выбирался двумя способами:

- путем визуальной оценки реконструированного изображения  $w_\alpha$  при разных значениях  $\alpha$  с выводом  $w_\alpha$  на экран и выбором такого значения  $\alpha$ , при котором зрительное восприятие изображения  $w_\alpha$  является "наилучшим"; при всей субъективности этого способа он является достаточно эффективным при реконструкции, в первую очередь, *реальных* искаженных изображений, главным образом, текстов, лиц, самолетов и других объектов, "знакомых" глазу (ср. [60, 91]);

- путем минимизации относительного среднеквадратического отклонения (СКО) изображения  $w_\alpha$  от точного изображения  $\bar{w}$ , или относительной среднеквадратической погрешности реконструкции изображения [86, 108, 112] (ср. [122])

$$\sigma_{\text{rel}}(\alpha) = \frac{\|w_\alpha - \bar{w}\|_{L_2}}{\|\bar{w}\|_{L_2}} = \frac{\sqrt{\sum_{j=1}^m \sum_{i=1}^n [(w_\alpha)_{ji} - \bar{w}_{ji}]^2}}{\sqrt{\sum_{j=1}^m \sum_{i=1}^n \bar{w}_{ji}^2}}; \quad (2.41)$$

специфика этого способа состоит в том, что  $\sigma_{\text{rel}}(\alpha)$  в нем можно вычислить лишь при реконструкции *модельного* изображения, когда  $\bar{w}$  известно. Этот способ напоминает, в частности, способ оценки качества различных изображений (на монетах и т.д.) путем их сравнения с подлинными образцами не только профессиональными экспертами, но и с помощью цифровой камеры и программного обеспечения [24, 25].

Вместо (2.41) можно использовать для оценки погрешности реконструкции изображения также выражения:  $\sigma_{\text{rel}}(\alpha) = \|w_\alpha - \bar{w}\|_{L_1} / \|\bar{w}\|_{L_1}$ ,  $\sigma(\alpha) = \|w_\alpha - \bar{w}\|_{L_2}$ ,  $\sigma(\alpha) = \|w_\alpha - \bar{w}\|_{L_1}$ ,  $\text{PSNR}(\alpha) = 10 \lg(\bar{w}_{\text{max}} \cdot m \cdot n / \|w_\alpha - \bar{w}\|_{L_2}^2)$ . Мы далее будем пользоваться в основном выражением (2.41) (типа отношения шум/сигнал), дающим удобную оценку относительной погрешности реконструкции изображения.

Для выбора параметра регуляризации  $\alpha$  можно использовать также следующие способы: способ невязки [22, с. 244], [102, с. 194] (для его реализации нужно знать погрешность  $\|\delta g\|$  измерения искаженного изображения  $g$ ); обобщенный принцип невязки [22, с. 242], [114, с. 75], [121, с. 13] (нужно знать  $\|\delta g\|$  и погрешность знания ФРТ  $\|\delta h\|$ ); способ псевдообратного оператора [22, с. 239], [100] (нужно знать  $\|\delta g\|$  и  $\|\delta h\|$ ); способ, основанный на характеристиках "протяженности" функции  $h$  [26] (нужно знать оценки расстояния между контрастными элементами, в частности, пиками изображения) и др. [22, с. 242–249], [115, с. 23], [158]. Эти способы особенно полезны при реконструкции реального изображения, когда точное изображение  $\bar{w}$  неизвестно. В этом случае имеем способ объективного численного выбора параметра  $\alpha$ , при котором восстановление изображения будет близким к оптимальному. А выбор  $\alpha$  может дать и количественную оценку погрешности (а значит, качества) реконструкции изображения, например, способом псевдообратного оператора (ср. с. 210).

В данной главе мы в основном будем использовать способ выбора  $\alpha$ , основанный на

минимизации относительной погрешности  $\sigma_{\text{rel}}(\alpha)$ , а также способ визуальной оценки реконструированного изображения  $w_\alpha$ . Обозначим через  $\alpha_{\text{opt}}$  то (оптимальное) значение  $\alpha$ , при котором  $\sigma_{\text{rel}}(\alpha) = \min$ , т.е.

$$\alpha_{\text{opt}} = \arg \min \sigma_{\text{rel}}(\alpha). \quad (2.42)$$

Что касается параметра  $K$ , входящего в параметрическую винеровскую фильтрацию (см. (2.40)), то ее значение выбирается аналогично способу выбора  $\alpha$ :

$$K_{\text{opt}} = \arg \min \sigma_{\text{rel}}(K), \quad (2.43)$$

где

$$\sigma_{\text{rel}}(K) = \frac{\sqrt{\sum_{j=1}^m \sum_{i=1}^n [(w_K)_{ji} - \bar{w}_{ji}]^2}}{\sqrt{\sum_{j=1}^m \sum_{i=1}^n \bar{w}_{ji}^2}}. \quad (2.44)$$

Для вычисления  $\sigma_{\text{rel}}$  согласно (2.41) была разработана собственная m-функция `rmsd.m` (относящаяся ко второй группе программ). Ниже дано ее краткое описание.

```
function [sigma_rel,gshift] = rmsd(g,w)
%Оценка относительного среднеквадратического отклонения (СКО)
%(Relative Mean Square Deviation) некоторого изображения g
%от другого изображения w и вычисление небольшого сдвига
%изображения g, дающего минимум СКО.
%В роли изображения g может выступать, в первую очередь,
%реконструированное изображение w_alpha, Wnr и т.п.,
%а также смазанное (типа gir в smearing), дефокусированное g1
%или/и зашумленное изображение gln, а в роли w - точное
%изображение.
%Входные параметры (аргументы):
%g - матрица MxN некоторого изображения (uint8 или double),
%w - матрица mxn другого изображения (uint8 или double),
%   например, g - восстановленное изображение, а w - точное
%   изображение.
%Условия: g и w должны быть одинакового типа (gray или RGB),
%   но необязательно одинакового класса и ширины,
%   левые края g и w должны совпадать (с точностью до 2 пкс),
%   количество строк M и m у g и w должно быть одинаковое,
%   а количество столбцов N и n может быть разное.
%Выходные параметры (аргументы):
%sigma_rel = rmsd - относительное СКО (double),
%gshift - немного сдвинутое изображение g, дающее минимум sigma_rel.
%Возможные варианты обращений к rmsd:
```

```

%[sigma_rel,gshift] = rmsd(g,w)
%или
%sigma_rel = rmsd(g,w)

```

**Текстовое изображение.** Сначала рассмотрим обработку *текста*. Текстовое изображение имеет ту специфику, что, во-первых, в нем обычно имеется хотя бы одна точка, которая при смазывании превращается в штрих, по которому можно оценить направление смаза и его величину  $\Delta$ , а во-вторых, качество реконструированного текста надежно оценивается визуально за счет его читаемости, поэтому параметры  $\alpha$  и  $K$  могут быть выбраны не только на основе минимизации  $\sigma_{\text{rel}}$  (см. (2.41)–(2.44)), но и визуально, что особенно важно, когда обрабатывается реальное изображение.

Рассмотрим исходное текстовое модельное изображение (фантом) `textp.bmp` — позитив и `textn.bmp` — его негатив, каждый размером  $618 \times 690 \times 3$  класса `uint8`. Необходимость использования помимо позитива еще и негатива обусловлена тем, что это может понизить погрешность реконструкции  $\sigma_{\text{rel}}$  (как будет показано ниже). Далее во всех головных программах `text1.m, \dots, text8.m` выполнено следующее:

```

RGBp=imread('textp.bmp'); %618x690x3 uint8 (позитив)
wp=rgb2gray(RGBp); %618x690 uint8
b=uint8(5); %background (фон) uint8
wp=wp-b; %введение фона
wp=double(wp); %618x690 double
%Fig2_10:
figure, subplot(1,2,1), imshow(uint8(wp)), title('wp 618x690')
RGBn=imread('textn.bmp'); %618x690x3 uint8 (негатив)
wn=rgb2gray(RGBn); %618x690 uint8
wn=wn+b; %введение фона
wn=double(wn); %618x690 double
subplot(1,2,2), imshow(uint8(wn)), title('wn 618x690')

```

На рис. 2.10 (см. `text1.m`) представлены серые варианты исходного изображения `wp` (позитив) и `wn` (негатив).

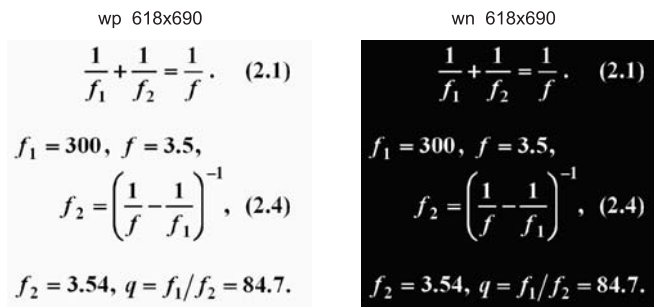


Рис. 2.10. `wp` — исходный текст, позитив;  
`wn` — исходный текст, негатив

**Прямая задача без зашумления.** Рассмотрим сначала *прямую задачу* — моделирование горизонтального смазывания изображения без зашумления (см. `text1.m`, `text2.m`, `text8.m`).

На рис. 2.11 (см. `text1.m`, `text2.m`) представлены результаты смазывания позитива `wp` и негатива `wn` с усечением (с помощью собственной *m*-функции `smearing` согласно (2.18)), а также результаты обращения полученных смазанных изображений (с помощью внешней *m*-функции `imcomplement`). При этом величина смаза равна  $\Delta = 20$  пкс, а угол смаза  $\theta = 0$  (горизонтальное смазывание).

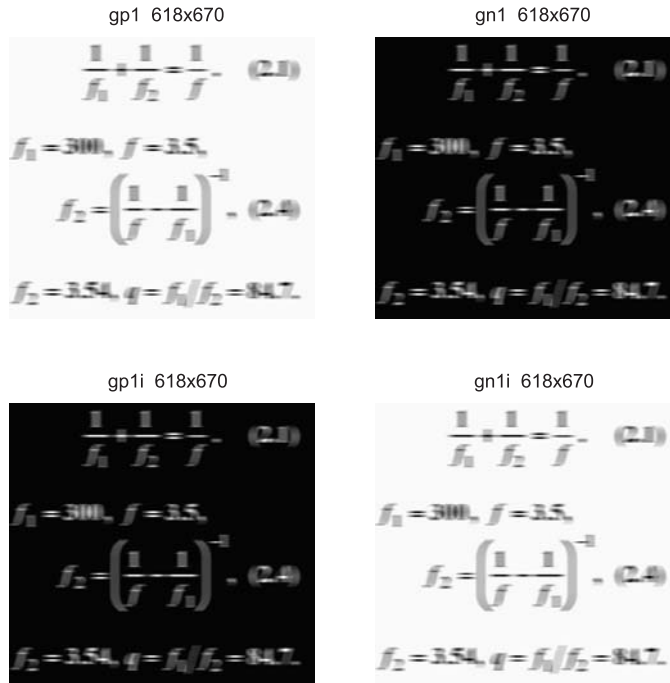


Рис. 2.11. Смазывание изображения с усечением

Изображения на рис. 2.11 были рассчитаны следующим образом:

```
gp1=smearing(wp,Delta,0,'truncation'); %смазывание позитива wp
gn1=smearing(wn,Delta,0,'truncation'); %смазывание негатива wn
gp1i=imcomplement(uint8(gp1)); %обращение изображения gp1
gn1i=imcomplement(uint8(gn1)); %обращение изображения gn1
```

На рис. 2.12 (см. `text1.m`, `text2.m`) — результаты смазывания позитива `wp` и негатива `wn` с размытием краев (с помощью функции `smearing` согласно (2.19), (2.20)) и результаты обращения смазанных изображений (с помощью функции `imcomplement`).

Изображения на рис. 2.12 были рассчитаны следующим образом:

```
gp2=smearing(wp,Delta,0,'diffusion'); %смазывание позитива wp
gn2=smearing(wn,Delta,0,'diffusion'); %смазывание негатива wn
```

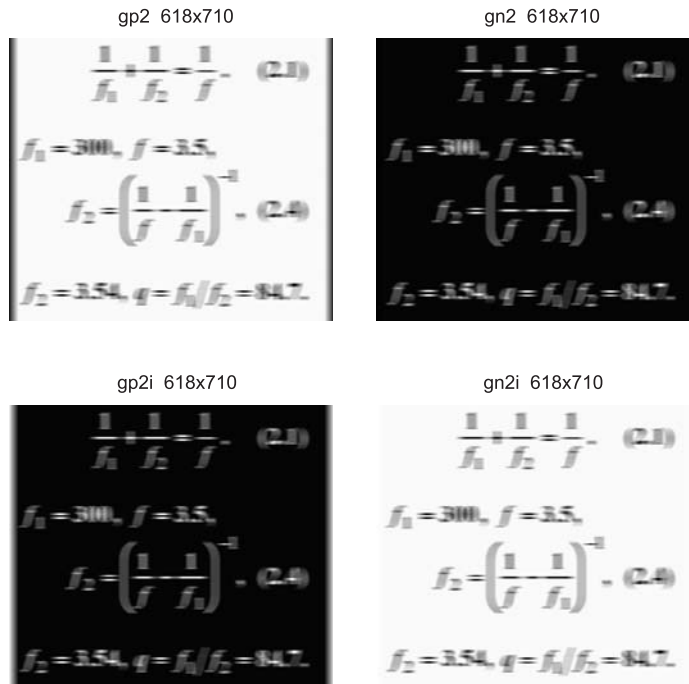


Рис. 2.12. Смазывание изображения с размытием краев

```
gp2i=imcomplement(uint8(gp2)); %обращение изображения gp2
gn2i=imcomplement(uint8(gn2)); %обращение изображения gn2
```

**Обратная задача без зашумления и с точной ФРТ.** Теперь обратимся к *обратной задаче* — реконструкции смазанных изображений также без зашумления и с точным знанием функции рассеяния точки, т.е. с точными значениями  $\Delta$  и  $\theta$ .

**Вариант 1.** На рис. 2.13 (см. `text1.m`) представлены зависимости относительной погрешности  $\sigma_{rel}(\alpha)$  (см. (2.41)) реконструкции смазанных с усечением изображений, приведенных на рис. 2.11 (кривые 1–6 на рис. 2.13) — вариант 1 реконструкции. Параметр регуляризации  $\alpha$  изменялся по логарифмическому закону с крупным шагом:

```

alpha_i=10^(-16.5); i=0;
while (alpha_i<=1e0+1e-4)
i=i+1; alpha(i)=alpha_i;
.....
alpha_i=alpha_i*10^0.5;
end %while

```

(2.45)

Затем для перехода на более мелкий шаг: `alpha_i=alpha_i*10^0.1` была выполнена интерполяция кривых многочленами Эрмита с помощью внешней `m`-функции `interp1(..., 'cubic')`. Это заметно сократило время расчета кривых на рис. 2.13.

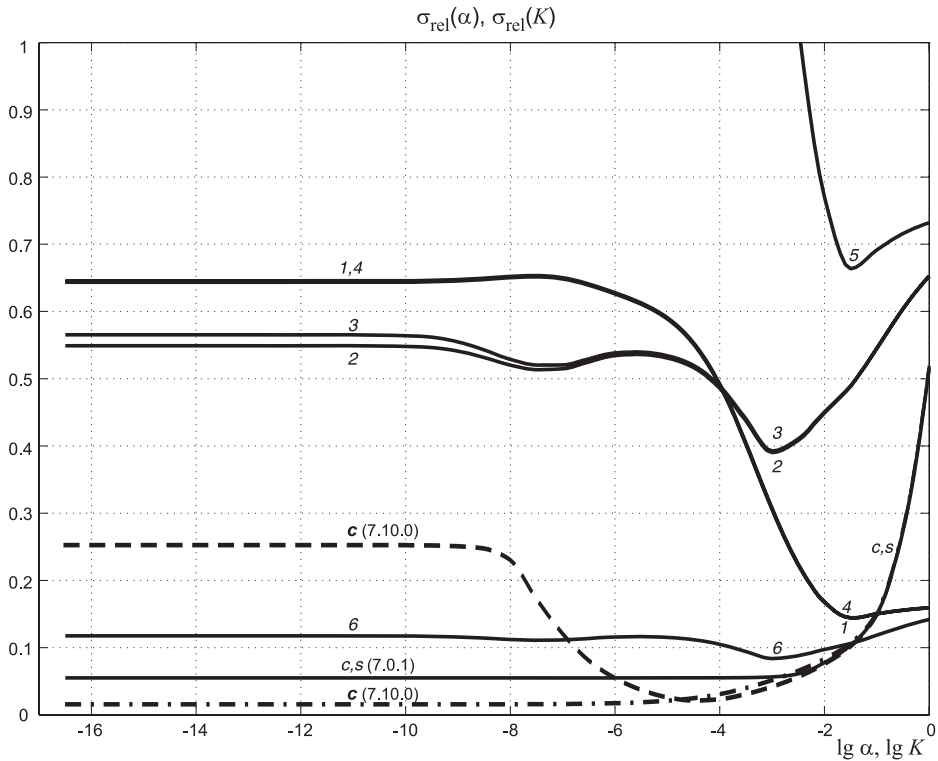


Рис. 2.13. Зависимости относительной погрешности  $\sigma_{rel}$  реконструкции смазанных с усечением изображений от  $\alpha$  (вариант 1,  $p = 1$ ) и от  $K$

Для сравнения с помощью внешних m-функций `fspecial` и `imfilter` было выполнено также смазывание изображения `wr`, а затем реконструкция смазанного изображения `mot` методом параметрической винеровской фильтрации (см. (2.40)) с помощью внешней m-функции `deconvnr`, причем параметр  $K$  изменялся по закону:

```

Ki=10^(-16.5); i=0;
while (Ki<=1e0+1e-4)
i=i+1; K(i)=Ki;
.....
Ki=Ki*10^0.5;
end %while
    
```

(2.46)

аналогичному (2.45). Была рассчитана зависимость  $\sigma_{rel}(K)$  согласно (2.44). При этом в функции `imfilter` использовалась опция `'circular'` (соответствует "граничному условию `periodic`"), а также `'symmetric'` (соответствует `reflective`). Кривые, соответствующие `'circular'` и `'symmetric'`, практически совпали (*c, s* на рис. 2.13). Кривые на рис. 2.13 были получены с помощью следующих операторов:



```

[m,n]=size(wp);
1 — w_alp=desmearingf(gp1,Delta,0,'var1',alpha_i,p,'mod',m,n);
   sig_rel1p(i)=rmsd(w_alp,wp);
2 — w_aln=desmearingf(gn1,Delta,0,'var1',alpha_i,p,'mod',m,n);
   sig_rel1n(i)=rmsd(w_aln,wn);
3 — w_alpi=desmearingf(gp1i,Delta,0,'var1',alpha_i,p,'mod',m,n);
   sig_rel1pi(i)=rmsd(w_alpi,wn);
4 — w_alni=desmearingf(gn1i,Delta,0,'var1',alpha_i,p,'mod',m,n);
   sig_rel1ni(i)=rmsd(w_alni,wp);
5 — w_alpii=imcomplement(uint8(w_alp));
   sig_rel1pii(i)=rmsd(w_alpii,wn);
6 — w_alnii=imcomplement(uint8(w_aln));
   sig_rel1nii(i)=rmsd(w_alnii,wp);
c,s — PSF=fspecial('motion',Delta,0);
      mot=imfilter(wp,PSF,'circular');
      K=alpha_i; Wnr=deconvwnr(uint8(mot),PSF,K);
      sig_relWnr(i)=rmsd(Wnr,wp);

```

**З а м е ч а н и е.** Выше уже было отмечено, что  $m$ -функция `deconvwnr` из различных подверсий версии MatLab 7 может давать (при малых значениях  $K$ ) заметно различные решения  $w_K$  и, значит, различные значения  $\sigma_{\text{rel}}(K)$ . Действительно, при обращении: `Wnr=deconvwnr(uint8(mot),PSF,K)`; где `mot` – типа `double`, получаем кривую `c,s(7.0.1)` в MatLab 7.0.1 и пунктирную кривую `c(7.10.0)` в MatLab 7.10.0 (см. рис. 2.13). Если же выполнить обращение: `Wnr=deconvwnr(mot,PSF,K)`; то в MatLab 7.0.1 получим кривую (не приведенную на рис. 2.13, например,  $\sigma_{\text{rel}}(10^{-4}) = 0.3702$ ), идущую значительно *выше* кривой `c,s(7.0.1)`, а в MatLab 7.10.0 получим штрих-пунктирную кривую `c(7.10.0)`, идущую значительно *ниже* пунктирной кривой `c(7.10.0)`. Это говорит о том, что нужно делать различные обращения к функции `deconvwnr` и выбирать наилучшее. Кроме того, это говорит о том, что задача реконструкции является некорректной (неустойчивой) и даже небольшие изменения в алгоритме и/или в программе могут привести к значительным изменениям при малых  $K$  в решении.

На рис. 2.14 (см. `text2.m`) представлены результаты реконструкций смазанных изображений, выполненных согласно варианту 1 при  $\alpha = \alpha_{\text{opt}}$ , причем  $\alpha_{\text{opt}}$  получено по кривым 1–6 рис. 2.13 согласно (2.42).

А на рис. 2.15 (см. `text8.m`) — результат реконструкции смазанного изображения методом параметрической фильтрации Винера при  $K = K_{\text{opt}}$ . Из рис. 2.13–2.15 видно, что реконструкция согласно варианту 1 (метод ПФ с регуляризацией и с усечением смазанного изображения) дает недостаточно хорошее восстановление изображений, лишь подвариант 6 (кривая 6 на рис. 2.13 и изображение `w_alnii` на рис. 2.14) дает удовлетворительный результат и это можно объяснить тем, что обрабатывался негатив `gn1`, у которого края являются черными и интенсивность гладко ниспадает до нуля к краям. Примерно такой же результат дает метод параметрической фильтрации Винера (кривая `c,s` на рис. 2.13 и изображение `Wnr` на рис. 2.15), причем винеровская

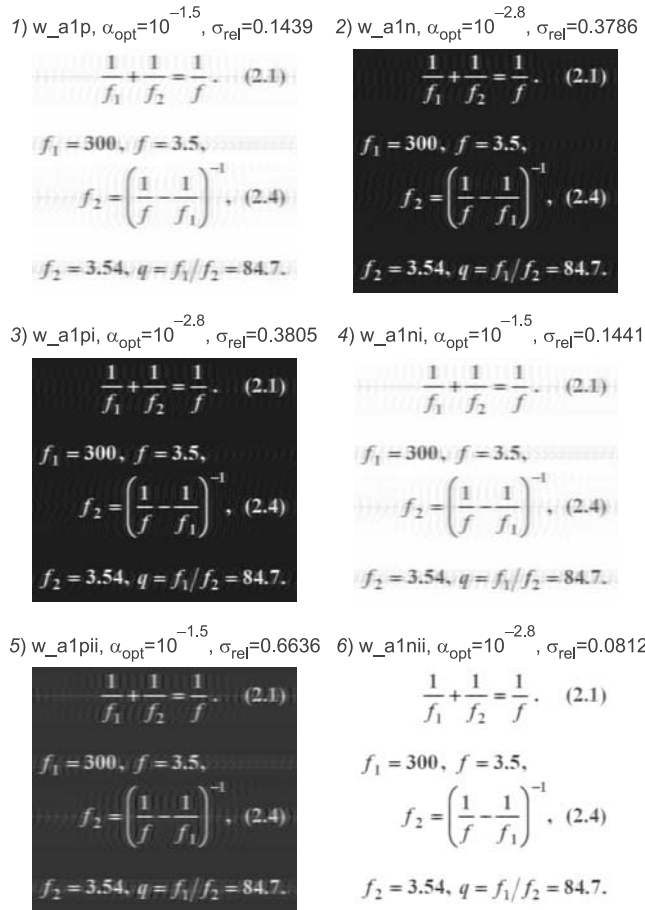


Рис. 2.14. Реконструкция смазанных с усечением изображений, вариант 1, при  $\alpha = \alpha_{opt}$ . Нумерация такая же, как на рис. 2.13 (1–6)

фильтрация имеет меньшее значение  $\sigma_{rel}$ , чем подвариант б, и визуально изображение  $w\_a1nii$  воспринимается хуже, чем изображение  $wnr$ .

**Вариант 2.** Далее на рис. 2.16 (см. `text1.m`) представлены зависимости относительной погрешности  $\sigma_{rel}(\alpha)$  (см. (2.41)) реконструкции смазанных с размытыми краями изображений, приведенных на рис. 2.12 (кривые 1–6 на рис. 2.16).

На рис. 2.16:

```
[m,n]=size(wp);
1 — w_a2p=desmearingf(gp2,Delta,0,'var2',alpha_i,p,'mod',m,n);
   sig_rel2p(i)=rmsd(w_a2p,wp);
2 — w_a2n=desmearingf(gn2,Delta,0,'var2',alpha_i,p,'mod',m,n);
   sig_rel2n(i)=rmsd(w_a2n,wn);
3 — w_a2pi=desmearingf(gp2i,Delta,0,'var2',alpha_i,p,'mod',m,n);
   sig_rel2pi(i)=rmsd(w_a2pi,wn);
4 — w_a2ni=desmearingf(gn2i,Delta,0,'var2',alpha_i,p,'mod',m,n);
```

<p style="text-align: center;">mot 618x690</p> $\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f} \quad (2.3)$ <p style="text-align: center;"><math>f_1 = 300, f = 3.5,</math></p> $f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1} \quad (2.4)$ <p style="text-align: center;"><math>f_2 = 3.54, q = f_1/f_2 = 84.7.</math></p>	<p style="text-align: center;">Wnr 618x690, <math>\sigma_{rel} = 0.0218</math></p> $\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f} \quad (2.1)$ <p style="text-align: center;"><math>f_1 = 300, f = 3.5,</math></p> $f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$ <p style="text-align: center;"><math>f_2 = 3.54, q = f_1/f_2 = 84.7.</math></p>
--	---

Рис. 2.15. Реконструкция смазанного изображения методом параметрической фильтрации Винера при  $K = K_{opt} = 10^{-4}$  (см. кривые  $c(7.10.0)$  на рис. 2.13)

```

sig_rel2ni(i)=rmsd(w_a2ni,wp);
5 — w_a2pii=imcomplement(uint8(w_a2p));
sig_rel2pii(i)=rmsd(w_a2pii,wn);
6 — w_a2nii=imcomplement(uint8(w_a2n));
sig_rel2nii(i)=rmsd(w_a2nii,wp);
c,s — как на рис. 2.13.
    
```

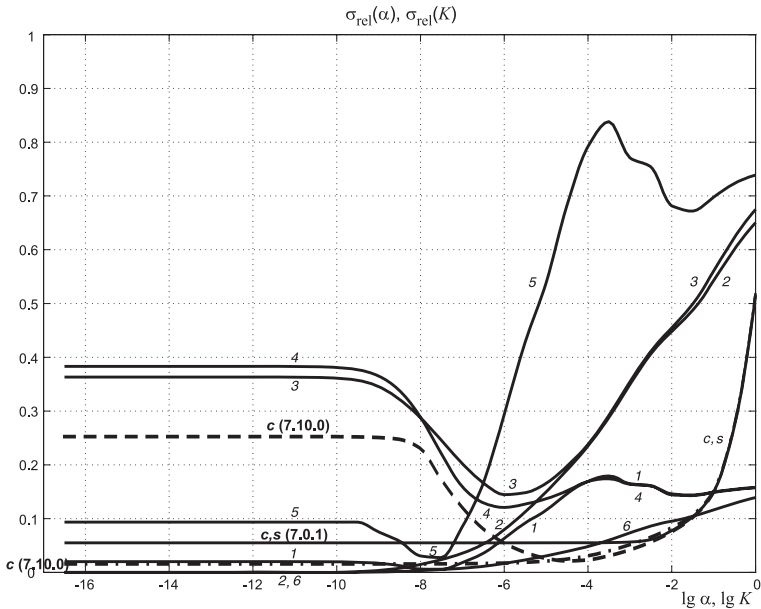


Рис. 2.16. Зависимости относительной погрешности  $\sigma_{rel}$  реконструкции смазанных изображений с размытыми краями от  $\alpha$  (вариант 2,  $p = 1$ ) и  $K$

На рис. 2.17 (см. `text2.m`) приведены результаты реконструкций смазанных изображений, выполненных согласно варианту 2 при  $\alpha = \alpha_{opt}$ , причем  $\alpha_{opt}$  получено по

кривым рис. 2.16 согласно (2.42) (а на рис. 2.15 для сравнения — результат реконструкции винеровской фильтрацией при  $K = K_{\text{opt}}$ ).

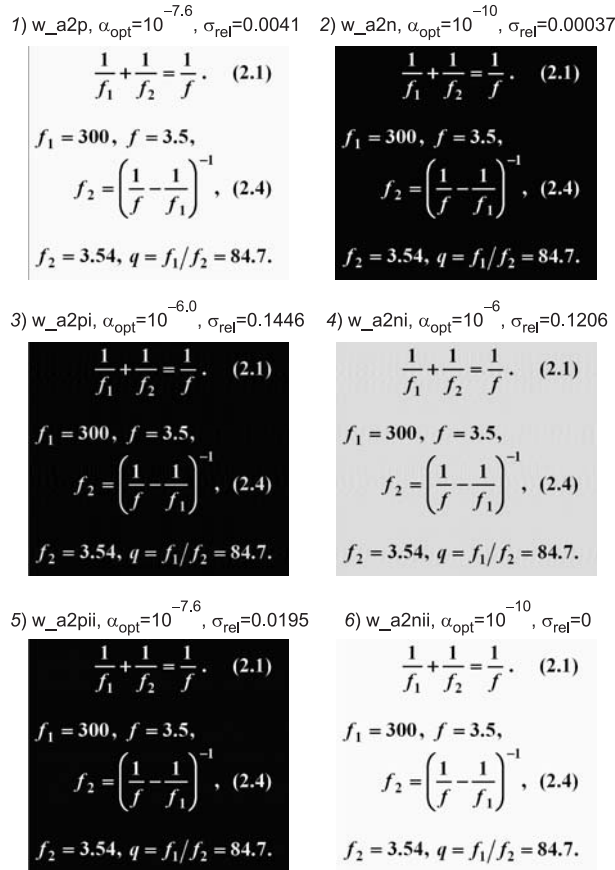


Рис. 2.17. Реконструкция смазанных с размытыми краями изображений; вариант 2,  $\alpha = \alpha_{\text{opt}}$ ; нумерация такая же, как на рис. 2.16 (1–6).

Размер каждого изображения  $618 \times 690$

Из рис. 2.16 и 2.17 видно, что реконструкция согласно варианту 2 (метод ПФ с регуляризацией и с размытием краев) дает несколько более точное восстановление изображений по сравнению с вариантом 1, особенно это касается подвариантов 1 и 6 (изображения  $w\_a2p$  и  $w\_a2nii$ ). При этом хотя подварианты 1 и 6 имеют меньшие значения  $\sigma_{\text{rel}}$ , чем подвариант  $c, s$  (винеровская фильтрация), тем не менее визуально изображения  $w\_a2p$  и  $w\_a2nii$  (см. рис. 2.17) воспринимаются практически так же, как  $w_{nr}$  (см. рис. 2.15), что можно объяснить особенностями физиологии зрения.

**Варианты 3, 4.** Перейдем к результатам реконструкции изображений, полученным методом конечных сумм (квадратур) с регуляризацией Тихонова (варианты 3–5, формулы (2.30)–(2.38), собственная  $m$ -функция `desmearingq`). Сначала остановимся на вариантах 3 и 4.

На рис. 2.18 (см. `text3.m`) представлены зависимости относительной погрешности

$\sigma_{\text{rel}}$  (см. (2.41)) реконструкции смазанных с усечением изображений, приведенных на рис. 2.11. Кривые 1–6 на рис. 2.18 получены путем применения варианта 3 (см. (2.32), (2.33), (2.37), (2.38), (2.41), (2.45)).

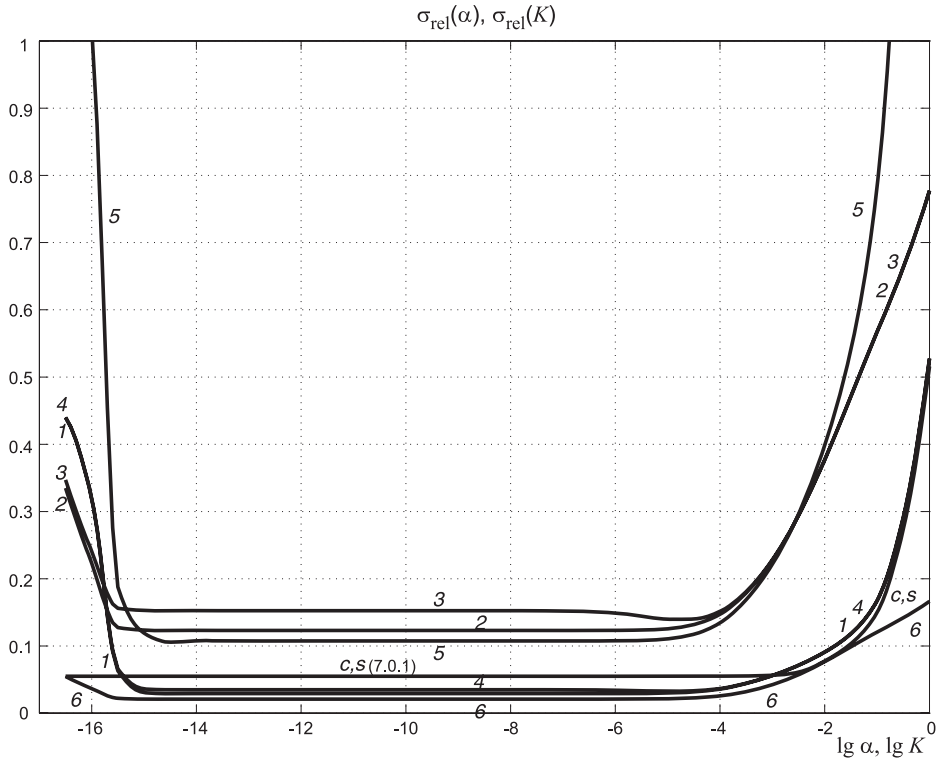


Рис. 2.18. Зависимости относительной погрешности  $\sigma_{\text{rel}}$  реконструкции смазанных с усечением изображений от  $\alpha$  (вариант 3) и от  $K$  (еще две кривые  $c(7.10.0)$  см. на рис. 2.13)

Кривые на рис. 2.18 были получены с помощью следующих операторов:

```
[m,n]=size(wp);
1 — w_a3p=desmearingq(gp1,Delta,0,'var3',alphi,'mod',m,n);
   sig_rel3p(i)=rmsd(w_a3p,wp);
2 — w_a3n=desmearingq(gn1,Delta,0,'var3',alphi,'mod',m,n);
   sig_rel3n(i)=rmsd(w_a3n,wn);
3 — w_a3pi=desmearingq(gp1i,Delta,0,'var3',alphi,'mod',m,n);
   sig_rel3pi(i)=rmsd(w_a3pi,wn);
4 — w_a3ni=desmearingq(gn1i,Delta,0,'var3',alphi,'mod',m,n);
   sig_rel3ni(i)=rmsd(w_a3ni,wp);
5 — w_a3pii=imcomplement(uint8(w_a3p));
   sig_rel3pii(i)=rmsd(w_a3pii,wn);
6 — w_a3nii=imcomplement(uint8(w_a3n));
   sig_rel3nii(i)=rmsd(w_a3nii,wp);
```

$c, s$  — как на рис. 2.13.

На рис. 2.19 (см. `text4.m`) приведены результаты реконструкции смазанных с усечением изображений, выполненных согласно варианту 3 при  $\alpha = \alpha_{\text{opt}}$ , причем значения  $\alpha_{\text{opt}}$  получены по кривым рис. 2.18 согласно (2.42) (а на рис. 2.15 приведен для сравнения результат реконструкции винеровской фильтрацией при  $K = K_{\text{opt}}$ ).

1)  $w\_a3p, \alpha_{\text{opt}}=10^{-7}, \sigma_{\text{rel}}=0.0291$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$$

$$f_1 = 300, f = 3.5,$$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$$f_2 = 3.54, q = f_1/f_2 = 84.7.$$

2)  $w\_a3n, \alpha_{\text{opt}}=10^{-7}, \sigma_{\text{rel}}=0.1231$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$$

$$f_1 = 300, f = 3.5,$$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$$f_2 = 3.54, q = f_1/f_2 = 84.7.$$

3)  $w\_a3pi, \alpha_{\text{opt}}=10^{-5}, \sigma_{\text{rel}}=0.1397$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$$

$$f_1 = 300, f = 3.5,$$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$$f_2 = 3.54, q = f_1/f_2 = 84.7.$$

4)  $w\_a3ni, \alpha_{\text{opt}}=10^{-5}, \sigma_{\text{rel}}=0.0326$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$$

$$f_1 = 300, f = 3.5,$$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$$f_2 = 3.54, q = f_1/f_2 = 84.7.$$

5)  $w\_a3pii, \alpha_{\text{opt}}=10^{-7}, \sigma_{\text{rel}}=0.1074$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$$

$$f_1 = 300, f = 3.5,$$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$$f_2 = 3.54, q = f_1/f_2 = 84.7.$$

6)  $w\_a3nii, \alpha_{\text{opt}}=10^{-5}, \sigma_{\text{rel}}=0.0210$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$$

$$f_1 = 300, f = 3.5,$$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$$f_2 = 3.54, q = f_1/f_2 = 84.7.$$

Рис. 2.19. Реконструкция смазанных с усечением изображений, вариант 3, при  $\alpha = \alpha_{\text{opt}}$ . Нумерация такая же, как на рис. 2.18 (1–6).

Размер каждого изображения  $618 \times 690$

Из рис. 2.18 и 2.19 видно, что реконструкция согласно варианту 3 (метод конечных сумм с регуляризацией и с усечением) дает заметно лучшие результаты, чем вариант 1 (рис. 2.13, 2.14), и худшие, чем вариант 2 (рис. 2.16, 2.17), и сравнимые с результатами винеровской фильтрации (рис. 2.15). Кривые на рис. 2.18 имеют заметно более глубокие и широкие минимумы, чем на рис. 2.13 и 2.16. Особенно это касается подвариантов 1, 4 и 6 на рис. 2.18, 2.19, имеющих меньшие значения  $\sigma_{\text{rel}}(\alpha_{\text{opt}})$ , чем  $\sigma_{\text{rel}}(K_{\text{opt}})$  в подварианте  $c, s$  (винеровская фильтрация).

На рис. 2.20 (см. `text3.m`) представлены зависимости относительной погрешности  $\sigma_{\text{rel}}(\alpha)$  (см. (2.41)) реконструкции смазанных с размытыми краями изображений, приведенных на рис. 2.12. Кривые 1–6 на рис. 2.20 получены путем применения варианта

4 (см. (2.34), (2.35), (2.37), (2.38), (2.41), (2.45)).

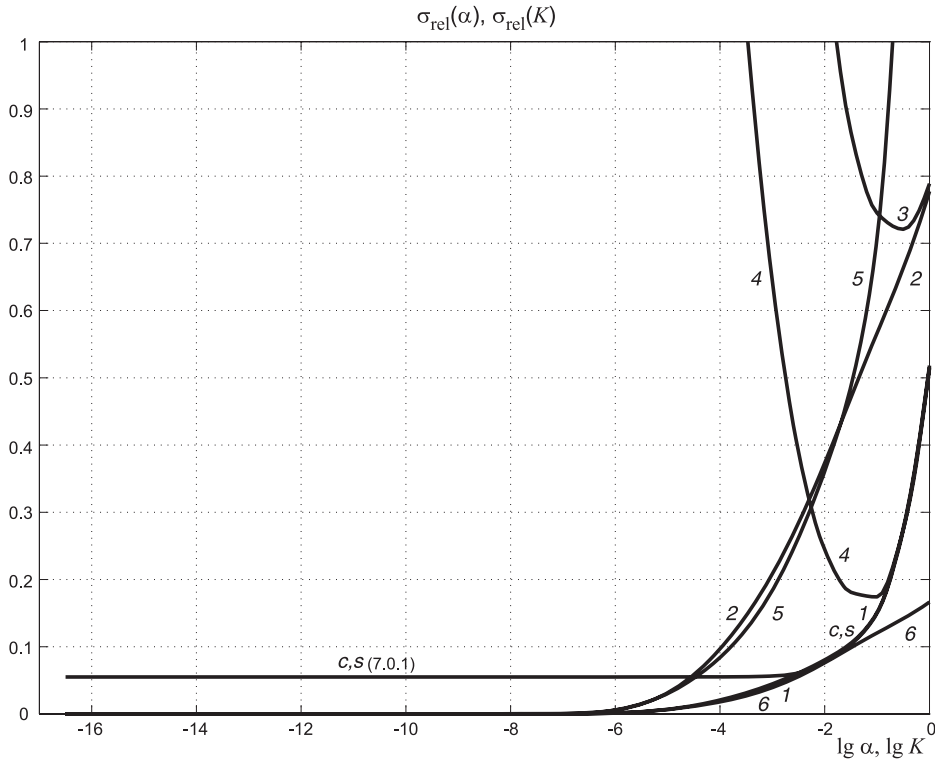


Рис. 2.20. Зависимости относительной погрешности  $\sigma_{\text{rel}}$  реконструкции смазанных изображений с размытыми краями от  $\alpha$  (вариант 4) и от  $K$  (еще две кривые  $c(7.10.0)$  см. на рис. 2.16)

Кривые на рис. 2.20 были получены с помощью следующих операторов:

```
[m,n]=size(wp);
1 — w_a4p=desmearingq(gp2,Delta,0,'var4',alpha_i,'mod',m,n);
   sig_rel4p(i)=rmsd(w_a4p,wp);
2 — w_a4n=desmearingq(gn2,Delta,0,'var4',alpha_i,'mod',m,n);
   sig_rel4n(i)=rmsd(w_a4n,wn);
3 — w_a4pi=desmearingq(gp2i,Delta,0,'var4',alpha_i,'mod',m,n);
   sig_rel4pi(i)=rmsd(w_a4pi,wn);
4 — w_a4ni=desmearingq(gn2i,Delta,0,'var4',alpha_i,'mod',m,n);
   sig_rel4ni(i)=rmsd(w_a4ni,wp);
5 — w_a4pii=imcomplement(uint8(w_a4p));
   sig_rel4pii(i)=rmsd(w_a4pii,wn);
6 — w_a4nii=imcomplement(uint8(w_a4n));
   sig_rel4nii(i)=rmsd(w_a4nii,wp);
```

На рис. 2.21 (см. text4.m) приведены реконструированные изображения согласно варианту 4 при  $\alpha = \alpha_{\text{opt}}$ , причем значения  $\alpha_{\text{opt}}$  получены по кривым рис. 2.20 согласно

(2.42) (а на рис. 2.15 приведен для сравнения результат реконструкции винеровской фильтрацией при  $K = K_{\text{opt}}$ ).

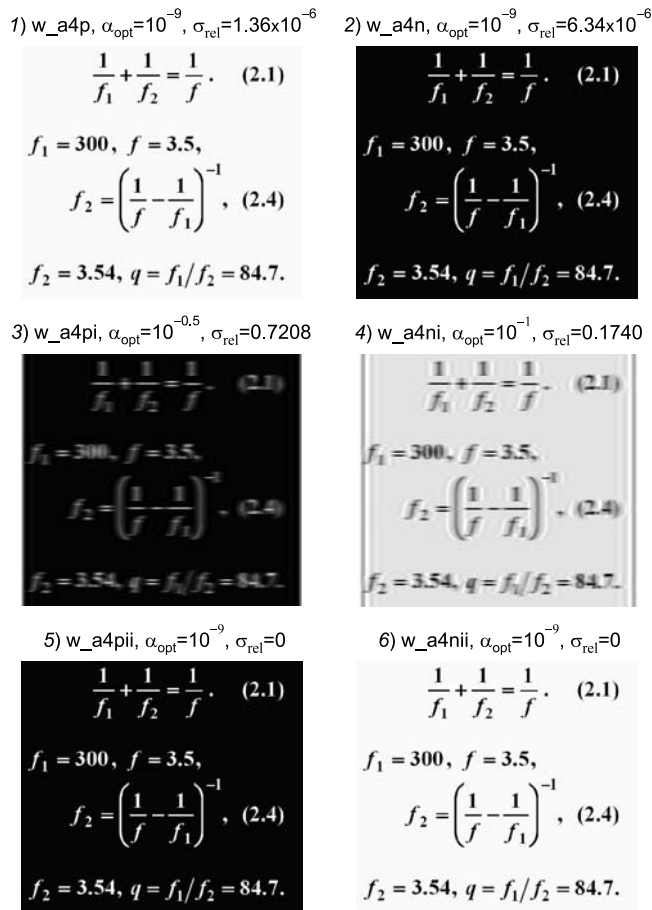


Рис. 2.21. Реконструкция смазанных с размытыми краями изображений, вариант 4, при  $\alpha = \alpha_{\text{opt}}$ . Нумерация такая же, как на рис. 2.20 (1–6).  
Размер каждого изображения  $618 \times 690$

Из рис. 2.20 и 2.21 видно, что реконструкция согласно варианту 4 (метод конечных сумм с регуляризацией и с размытием) дает значительно более точные результаты, чем варианты 1, 2 и даже 3 (рис. 2.13, 2.14, 2.16–2.19). Кривые  $\sigma_{\text{rel}}(\alpha)$  1, 2, 5 и 6 на рис. 2.20 опускаются практически до нуля при  $\lg \alpha \lesssim -7$ , в результате чего подварианты 1, 2, 5 и 6 на рис. 2.21 дают практически точное восстановление как позитива **wp**, так и негатива **wn** (см. рис. 2.10).

При этом отметим, что подвариант 4 и особенно подвариант 3 (на рис. 2.20 и 2.21) дают очень грубое восстановление:  $\sigma_{\text{rel}}(\alpha_{\text{opt}}) \approx 21\%$  и  $79\%$  соответственно. Это можно объяснить тем, что в подварианте 4 обрабатывалось изображение **gn2i** (см. рис. 2.12) с белыми неразмытыми краями. В результате интенсивность  $g$  на изображении **gn2i** скачкообразно падает к нулю на краях и, как отмечалось выше, при решении



обратной задачи это ведет к появлению эффекта Гиббса. Еще сильнее этот эффект проявляется в подварианте 3, в котором обрабатывалось изображение `gp2i` (см. рис. 2.12), у которого интенсивность  $g$  при приближении к левому и правому краям даже нарастает ("от черного к белому").

Данные результаты позволяют сделать следующие предварительные выводы.

1) Для повышения точности реконструкции смазанное изображение должно иметь (или нужно искусственно добавить) размытые края с гладким падением интенсивности к нулю типа изображений `gp2` и `gn2` на рис. 2.12, но не с постоянством высокой интенсивности на краях, как на изображениях `gp1`, `gn1i` на рис. 2.11 и `gn2i` на рис. 2.12, тем более, не с нарастанием интенсивности к краям, как на изображении `gp2i` на рис. 2.12.

2) При моделировании смазывания (прямая задача) и при реконструкции изображения (обратная задача) целесообразно вместо обычно используемого преобразования Фурье (см. [2], [35, с. 384–387], а также варианты 1 и 2) использовать операции суммирования (2.18)–(2.20) и метод конечных сумм (квадратур) с регуляризацией (варианты 3–5). Эта рекомендация связана с тем, что операция суммирования более адекватна физической природе смазывания изображения, чем операция преобразования Фурье.

**Использование функции `edgetaper.m`.** Рассмотрим вопрос об использовании внешней  $m$ -функции `edgetaper`. Дело в том, что в книге [36, с. 185] рекомендуется применять функцию `edgetaper` в случае, когда реконструированное изображение имеет искажения типа "звон" (эффект Гиббса). Обращение к этой функции имеет вид:  $J = \text{edgetaper}(I, \text{PSF})$ , где  $I$  — исходное (смазанное) изображение,  $\text{PSF}$  — функция рассеяния точки, а  $J$  — изображение, края которого размыты с помощью функции  $\text{PSF}$ , выступающей в роли автокорреляционной функции, а остальная (бóльшая) часть равна  $I$ .

Для проверки эффективности использования функции `edgetaper` были реализованы следующие операторы (см. `text7.m`):

```
[m,n]=size(wp);
gp1=smearing(wp,Delta,0,'truncation'); %смазывание позитива wp с усечением
PSF=fspecial('motion',Delta,0); %ФПТ
gpe=edgetaper(gp1,PSF); %размытие краев у изображения gp1
%Реконструкция, вариант 3:
w_a3pe=desmearingq(gpe,Delta,0,'var3',1e-7,'mod',m,n);
gp2=smearing(wp,Delta,0,'diffusion'); %смазывание позитива wp с размытием
%Реконструкция, вариант 4:
w_a4p=desmearingq(gp2,Delta,0,'var4',1e-9,'mod',m,n);
```

На рис. 2.22 представлены результаты такой проверки. Видно, что функция `edgetaper` не изменяет ширину изображения (`gpe` получается такой же ширины, как и `gp1`) и выполняет очень незначительное размытие краев изображения `gp1`. В то же время функция `smearing` делает значительно большее размытие краев изображения (см. `gp2` на рис. 2.12). В результате реконструкция изображения `gpe` дает практически такой же результат (см. `w_a3pe` на рис. 2.22), как и реконструкция изображения `gp1`

(см. `w_a3p` на рис. 2.19), т.е. применение функции `edgetaper` практически не изменило результат как в прямой, так и в обратной задаче. В то же время реконструкция изображения `gp2` позволила получить изображение `w_a4p` (см. рис. 2.22) со значительно меньшей погрешностью  $\sigma_{\text{rel}}$ , чем у изображения `w_a3pe` (см. рис. 2.22). Таким образом, можно сделать (предварительное) заключение, что использование функции `edgetaper` является значительно менее эффективным, чем использование функции `smearing(..., 'diffusion')`.

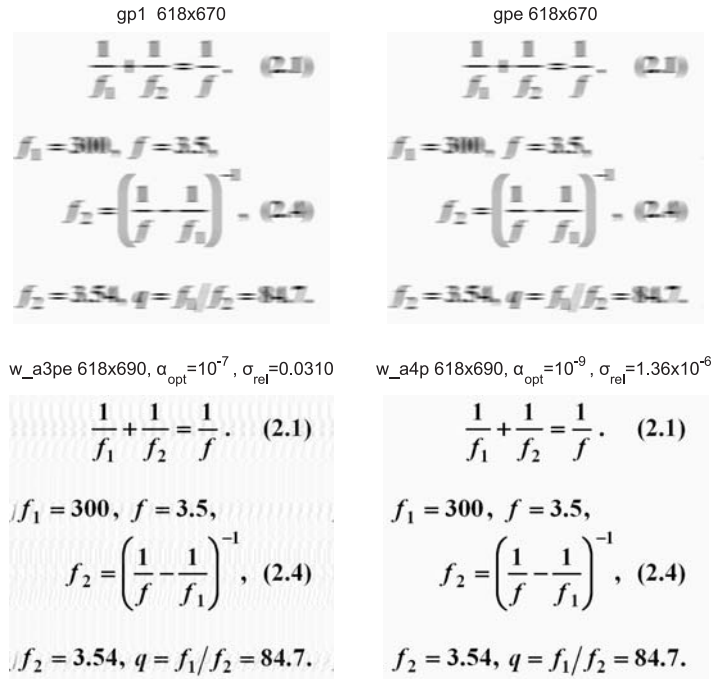


Рис. 2.22. Использование m-функции `edgetaper`

**Вариант 5.** Рассмотрим также вариант 5 — реконструкцию по схеме двойного усечения согласно (2.36), (2.37), (2.38), (2.41), (2.45). На рис. 2.23 (см. `text5.m`) представлены зависимости относительной погрешности  $\sigma_{\text{rel}}(\alpha)$  (см. (2.41)) реконструкции смазанных с усечением изображений, приведенных на рис. 2.11. Кривые на рис. 2.23 получены путем решения варианта 5 с помощью следующих операторов:

```
[m,n]=size(wp);
1 — w_a5p=desmearingq(gp1,Delta,0,'var5',alpha_i,'mod',m,n);
   sig_rel5p(i)=rmsd(w_a5p,wp);
2 — w_a5n=desmearingq(gn1,Delta,0,'var5',alpha_i,'mod',m,n);
   sig_rel5n(i)=rmsd(w_a5n,wn);
3 — w_a5pi=desmearingq(gp1i,Delta,0,'var5',alpha_i,'mod',m,n);
   sig_rel5pi(i)=rmsd(w_a5pi,wn);
4 — w_a5ni=desmearingq(gn1i,Delta,0,'var5',alpha_i,'mod',m,n);
   sig_rel5ni(i)=rmsd(w_a5ni,wp);
```

```

5 — w_a5pii=imcomplement(uint8(w_a5p));
   sig_rel5pii(i)=rmsd(w_a5pii,wn);
6 — w_a5nii=imcomplement(uint8(w_a5n));
   sig_rel5nii(i)=rmsd(w_a5nii,wp);
s — PSF=fspecial('motion',Delta,0);
   mot=imfilter(wp,PSF,'symmetric');
   K=alpha; Wnr=deconvwnr(uint8(mot),PSF,K);
   sig_relWnr(i)=rmsd(Wnr,wp);

```

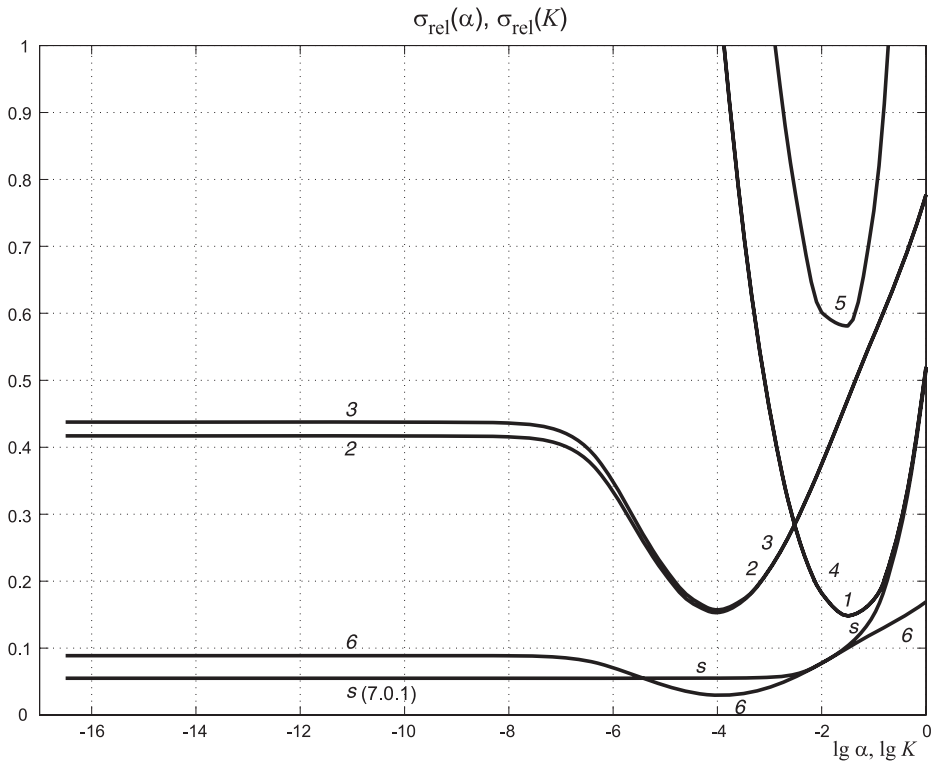


Рис. 2.23. Зависимости относительной погрешности  $\sigma_{rel}$  реконструкции смазанных с усечением изображений от  $\alpha$  (вариант 5) и от  $K$  (еще две кривые  $s(7.10.0)$  см. на рис. 2.13)

На рис. 2.24 (см. `text6.m`) приведены изображения, реконструированные согласно варианту 5 при  $\alpha = \alpha_{opt}$ , причем значения  $\alpha_{opt}$  получены по кривым рис. 2.23 согласно (2.42). На рис. 2.15 приведено для сравнения изображение  $Wnr$ , реконструированное винеровской фильтрацией при  $K = K_{opt}$ . Из рис. 2.23 и 2.24 видно, что вариант 5 выполняет реконструкцию изображений хуже, чем вариант 4 (см. рис. 2.20 и 2.21) и даже вариант 3 (рис. 2.18 и 2.19). Это можно объяснить тем, что в варианте 5 имеет место несоответствие прямой задачи (2.18) и обратной задачи (2.36), а именно, в прямой задаче используются значения  $w$  правого края изображения шириной  $\Delta$  (см.

рис. 2.8), в обратной задаче они исключаются из рассмотрения, т.е. полагается, что  $\widehat{w}_{j,k} = 0$  при  $k = n - \Delta + 1, \dots, n$  (см. (2.36) и рис. 2.8).

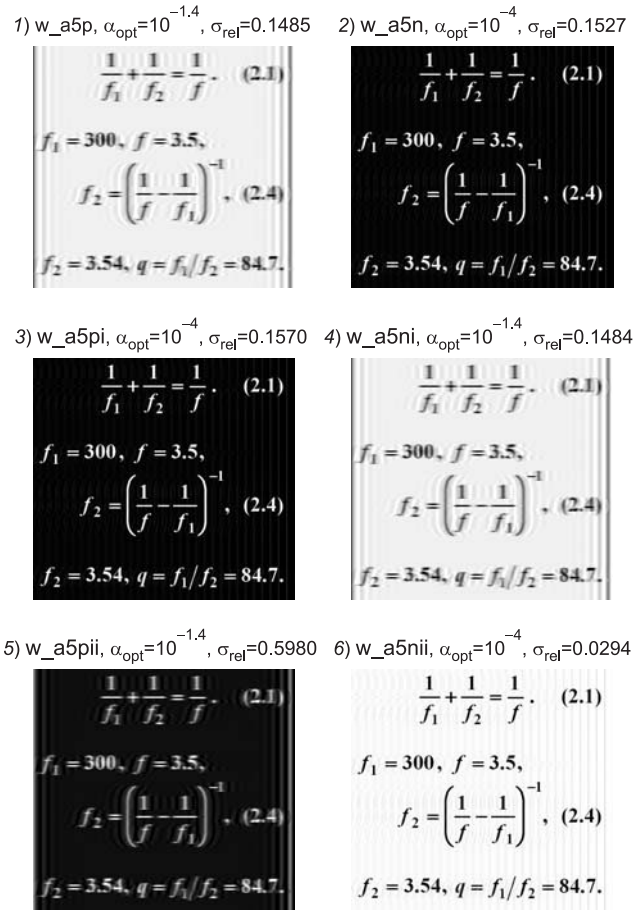


Рис. 2.24. Реконструкция смазанных с усечением изображений, вариант 5, при  $\alpha = \alpha_{\text{opt}}$ . Нумерация такая же, как на рис. 2.23 (1–6). Размер каждого изображения  $618 \times 670$

Итак, в дополнение к выводам 1)–2), сделанным выше (на стр. 113), можно сделать следующие в ы в о д ы .

3) Среди рассмотренных вариантов наилучшие результаты по реконструкции смазанных (незашумленных) изображений дает вариант 4 (см. рис. 2.20 и 2.21) и, в первую очередь, его подварианты 1 и 6, если под окончательным восстановленным изображением подразумевать позитив (подварианты 1 и 6 на рис. 2.21).

4) Применение внешней m-функции `edgetaper` не повышает точность реконструкции (см. изображения `gre` и `w_a3pe` на рис. 2.22). Эффективнее оказывается использование собственной m-функции `smearing(..., 'diffusion')` (см. изображения `gp2` на рис. 2.12 и `w_a4p` на рис. 2.22).

5) Метод параметрической фильтрации Винера с использованием так называемых "граничных условий" (BCs) дает заметно более худший результат реконструкции (см. кривые *c*, *s* на рис. 2.18 и 2.20 и изображение *Wnr* на рис. 2.15), чем вариант 4 (см. кривые *1* и *б* на рис. 2.20 и изображения *w\_a4p* и *w\_a4ni* на рис. 2.21) и даже вариант 3 (см. кривые *1* и *б* на рис. 2.18 и изображения *w\_a3p* и *w\_a3ni* на рис. 2.19). Напомним, что в вариантах 3 и 4 вместо искусственных BCs использован *прием усечения*, а в варианте 4 дополнительно *прием размытия*.

**Зашумленность смазанного изображения и погрешность ФРТ.** Теперь рассмотрим вопрос о влиянии зашумленности смазанных изображений на их восстановление. Рассмотрим следующий п р и м е р. На смазанное изображение *gp2* (см. рис. 2.12) был наложен гауссовый, а также импульсный (двоичный [142, с. 311]) шум. *Гауссовый шум* обычно имитирует непрерывный случайный процесс, например, туман между снимаемым объектом и устройством регистрации изображения (УРИ). Особенность гауссова шума состоит в том, что он присутствует в каждом пикселе изображения. *Импульсный же шум* имитирует снежинки, капли дождя, пыль в среде и т.д. и показателем импульсного шума является доля зашумленных пикселей на изображении. Спрашивается: как реконструировать смазанное и зашумленное изображение? Может быть д в а п у т и: а) реконструкция методом конечных сумм с регуляризацией Тихонова (лучше всего, вариантом 4) в предположении, что этот метод сам не только редуцирует смазывание, но и подавит шум; б) сначала — *предшествующая фильтрация шума* такими специальными методами, как адаптивная фильтрация Винера [35, с. 357–360], [45, с. 553] или медианная фильтрация (впервые предложенная Тьюки в 1974 г.) [36, с. 119], [45, с. 550], [129, с. 68], а затем — реконструкция смазанного (но очищенного от шума) изображения 4-м вариантом.

При этом в случае б) фильтрация шумов специальными методами может быть выполнена и после устранения смазывания изображения (назовем ее *последующей фильтрацией шума*). Предшествующая фильтрация шума (до устранения смазывания) целесообразна в случае, когда подвижный объект смазан за счет его движения, а шум (пыль в среде перед объектом и т.д.) не смазан. Если же смазывание произошло из-за движения УРИ (фотоаппарата и т.д.), то будет смазан и объект, и шум и в этом случае целесообразна последующая фильтрация шума (после устранения смазывания).

Кроме того, помехи и шумы можно разделить на две различные группы. К первой группе следует отнести внешние шумы (туман, пыль в среде и т.д.), которые накладываются на сигнал до его вхождения в УРИ. Ко второй группе следует отнести шумы, возникающие в самом УРИ (пыль на ПЗС-матрице, неоднородности чувствительности сенсоров матрицы, шумы квантования, электромагнитные помехи и т.д.). Различие этих групп состоит в том, что первый вид шумов можно или фильтровать (устранять), или восстанавливать так же, как полезные данные на изображении, а второй вид шумов подлежит только устранению.

Важным является также вопрос о влиянии погрешности знания функции рассеяния точки (в данном случае, погрешности  $\Delta$ ) на результат реконструкции (ср. [26–28]).

Для разрешения этих вопросов были записаны команды (см. `text3err.m`):

```

Delta=20; %Точное значение Delta
%Смазывание изображения с размытыми краями:
gp2=smearing(wp,Delta,0,'diffusion');
randn('state',0); %установка генератора случ. чисел в нач. состоянии
gp2gauss1=normnoise(gp2,1); %добавление гауссова 1%-ного шума к gp2
gp2gauss3=normnoise(gp2,3); %добавление гауссова 3%-ного шума к gp2
%Dобавление импульсного шума:
gp2imp=double(innoise(uint8(gp2),'salt & pepper',0.03));
%Фильтрация гауссова 3%-ного шума адаптивным фильтром Винера:
Wien_gauss3=wiener2(gp2gauss3,[5 5]);
%Фильтрация импульсного шума медианным фильтром:
med_imp=medfilt2(gp2imp,[5 5]);
%Точная функция рассеяния точки (ФРТ):
PSF=fspecial('motion',Delta,0);
PSF1=fspecial('motion',Delta+2,0); %ошибочная ФРТ

```

На рис. 2.25 (см. `text3err.m`, `text4err.m`) представлены полученные изображения `gp2`, `gp2gauss1`, `gp2gauss3`, `gp2imp`, `Wien_gauss3` и `med_imp`.

**З а м е ч а н и е.** Гауссовый шум обычно лучше фильтруется адаптивным фильтром Винера, а импульсный шум — медианной фильтрацией [45], поэтому к изображению `gp2gauss3` была применена функция `wiener2`, а к изображению `gp2imp` — функция `medfilt2`. Отметим, что функции `wiener2.m` и `medfilt2.m` фильтруют только серые шумы и не фильтруют цветные шумы. Далее (на стр. 172) приведены разработанные нами функции `wiener2с.m` и `medfilt2с.m`, фильтрующие как серые, так и цветные (RGB) шумы.

Затем в цикле по  $\alpha$  (2.45) были рассчитаны относительные погрешности реконструкции  $\sigma_{rel}(\alpha)$  путем использования варианта 4, представленные на рис. 2.26 (см. `text3err.m`).

Для расчета кривых на рис. 2.26 было выполнено следующее (`[m,n]=size(wp)`):

1 — реконструкция смазанного изображения с размытыми краями без шума, но с ошибочным значением  $\Delta$ :

```

w_a4perr=desmearingq(gp2,Delta+2,0,'var4',alpha,'mod',m,n);
sig_rel4perr(i)=rmsd(w_a4perr,wp);

```

2 — реконструкция смазанного и зашумленного 1%-ным гауссовым шумом изображения с размытыми краями:

```

w_a4pgauss1=desmearingq(gp2gauss1,Delta,0,'var4',alpha,'mod',m,n);
sig_rel4pgauss1(i)=rmsd(w_a4pgass1,wp);

```

3 — реконструкция смазанного и зашумленного 3%-ным гауссовым шумом изображения с размытыми краями:

```

w_a4pgauss3=desmearingq(gp2gauss3,Delta,0,'var4',alpha,'mod',m,n);
sig_rel4pgauss3(i)=rmsd(w_a4pgass3,wp);

```

4 — реконструкция смазанного и зашумленного импульсным шумом изображения (доля зашумленных пикселей, или плотность шума  $d = 0.03$ ) с размытыми краями:

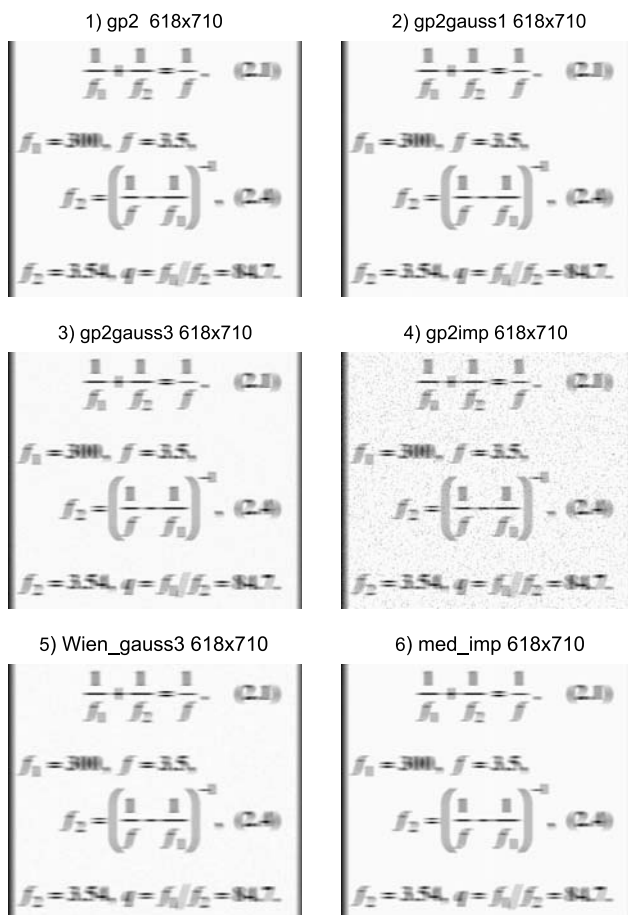


Рис. 2.25. Изображения со смазыванием, размытием, зашумленностью и фильтрацией:

- 1) gp2 — смазанное изображение; 2) к gp2 добавлен 1%-ный гауссовый шум;
- 3) к gp2 добавлен 3%-ный гауссовый шум; 4) к gp2 добавлен импульсный шум;
- 5) изображение gp2gauss3 отфильтровано адаптивным фильтром Винера;
- 6) изображение gp2imp отфильтровано медианным фильтром

```
w_a4pimp=desmearingq(gp2imp,Delta,0,'var4',alpha,'mod',m,n);
```

```
sig_rel4pimp(i)=rmsd(w_a4pimp,wp);
```

5 — реконструкция смазанного изображения с размытыми краями, у которого предварительно отфильтрован 3%-ный гауссовый шум адаптивной фильтрацией Винера:

```
w_a4pWgauss3=desmearingq(Wien_gauss3,Delta,0,'var4',alpha,'mod',m,n);
```

```
sig_rel4pWgauss3(i)=rmsd(w_a4pWgauss3,wp);
```

6 — реконструкция смазанного изображения с размытыми краями, у которого предварительно отфильтрован импульсный шум медианной фильтрацией:

```
w_a4pmimp=desmearingq(med_imp,Delta,0,'var4',alpha,'mod',m,n);
```

```
sig_rel4pmimp(i)=rmsd(w_a4pmimp,wp);
```

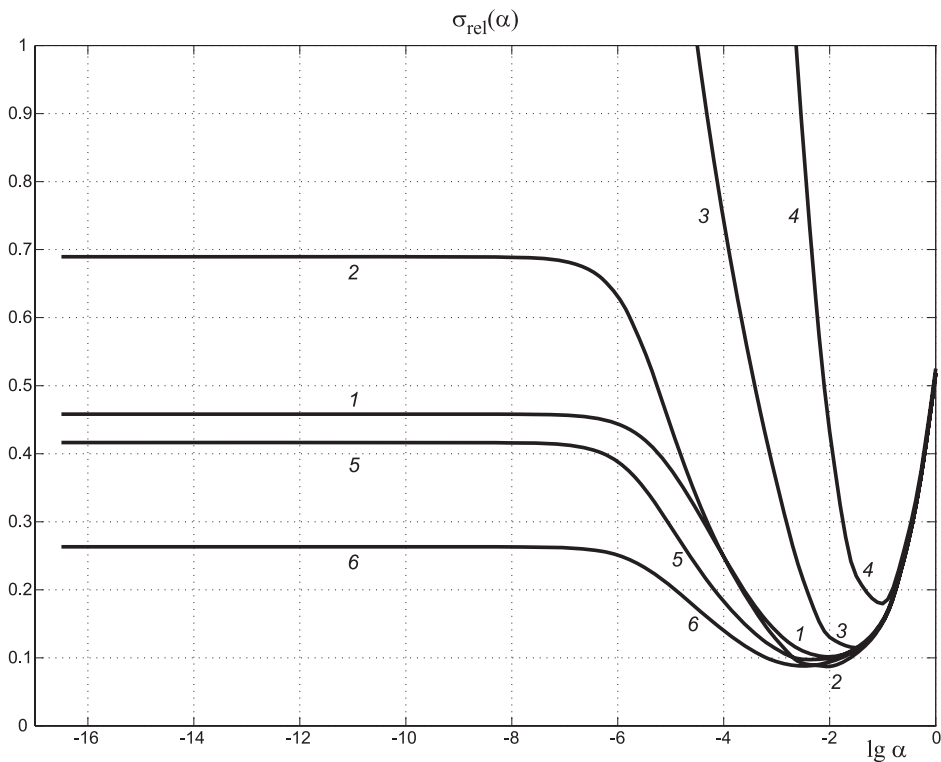


Рис. 2.26. Зависимости относительной погрешности  $\sigma_{rel}(\alpha)$  реконструкции смазанных с размытыми краями и зашумленных изображений (вариант 4)

На рис. 2.27 (см. `text4err.m`) приведены изображения, реконструированные согласно варианту 4 при  $\alpha = \alpha_{opt}$ , причем значения  $\alpha_{opt}$  получены по кривым 1–6 рис. 2.26 согласно (2.42).

Для получения изображений на рис. 2.27 были выполнены следующие операторы (`[m,n]=size(wp)`):

```
1 — w_a4perr=desmearingq(gp2,Delta+2,0,'var4',1e-2,'mod',m,n);
```

```
   %без шума, но с ошибочным Delta
```



1) w\_a4perr,  $\alpha=10^{-2}$ ,  $\sigma_{rel}=0.1013$     2) w\_a4pgauss1,  $\alpha=10^{-2.2}$ ,  $\sigma_{rel}=0.0855$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f} \quad (2.1)$$

$f_1 = 300, f = 3.5,$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$f_2 = 3.54, q = f_1/f_2 = 84.7.$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f} \quad (2.1)$$

$f_1 = 300, f = 3.5,$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$f_2 = 3.54, q = f_1/f_2 = 84.7.$

3) w\_a4pgauss3,  $\alpha=10^{-1.5}$ ,  $\sigma_{rel}=0.1147$     4) w\_a4pimp,  $\alpha=10^{-1}$ ,  $\sigma_{rel}=0.1797$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f} \quad (2.1)$$

$f_1 = 300, f = 3.5,$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$f_2 = 3.54, q = f_1/f_2 = 84.7.$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f} \quad (2.1)$$

$f_1 = 300, f = 3.5,$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$f_2 = 3.54, q = f_1/f_2 = 84.7.$

5) w\_a4pWgauss3,  $\alpha=10^{-2.3}$ ,  $\sigma_{rel}=0.0960$     6) w\_a4pmimp,  $\alpha=10^{-2.6}$ ,  $\sigma_{rel}=0.0876$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f} \quad (2.1)$$

$f_1 = 300, f = 3.5,$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$f_2 = 3.54, q = f_1/f_2 = 84.7.$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f} \quad (2.1)$$

$f_1 = 300, f = 3.5,$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$f_2 = 3.54, q = f_1/f_2 = 84.7.$

Рис. 2.27. Реконструкция смазанных изображений с погрешностями и шумами методом конечных сумм с регуляризацией Тихонова при  $\alpha = \alpha_{opt}$  (вариант 4).

Размер каждого изображения  $618 \times 690$  кроме w\_a4perr размера  $618 \times 688$

```
2 — w_a4pgauss1=desmearingq(gp2gauss1,Delta,0,'var4',10^(-2.2),'mod',m,n);
   %с 1%-м гауссовым шумом
3 — w_a4pgauss3=desmearingq(gp2gauss3,Delta,0,'var4',10^(-1.5),'mod',m,n);
   %с 3%-м гауссовым шумом
4 — w_a4pimp=desmearingq(gp2imp,Delta,0,'var4',1e-1,'mod',m,n);
   %с импульсным шумом
5 — w_a4pWgauss3=desmearingq(Wien_gauss3,Delta,0,'v4',10^(-2.3),'m',m,n);
   %с предварительной фильтрацией 3%-ного гауссова шума
   %адаптивным фильтром Винера
6 — w_a4pmimp=desmearingq(med_imp,Delta,0,'var4',10^(-2.6),'mod',m,n);
   %с предварительной фильтрацией импульсного шума
   %медианным фильтром
```

На основе анализа рис. 2.25–2.27 можно сделать следующие предварительные

в ы в о д ы (дополняющие выводы 1)–5), сделанные выше на стр. 113 и 116).

6) Сравнение кривой 1 на рис. 2.20 и изображения 1 на рис. 2.21 (реконструкция с точной ФРТ) с кривой 1 на рис. 2.26 и изображением 1 на рис. 2.27 (реконструкция с ошибочной ФРТ) показывает, что использование ФРТ с погрешностью (в нашем случае использование  $\tilde{\Delta} = 22$  пкс вместо  $\Delta = 20$  пкс, т.е. с погрешностью  $\approx 10\%$ ) ведет к заметному повышению погрешности реконструкции  $\sigma_{\text{rel}}(\alpha_{\text{opt}})$  методом конечных сумм и регуляризации Тихонова, вариант 4 ( $\sigma_{\text{rel}} \approx 11\%$ , см. рис. 2.27, 1).

7) Сравнение кривой 3 на рис. 2.26 и изображения 3 на рис. 2.27 (реконструкция смазанного и зашумленного 3%-ным гауссовым шумом изображения) с кривой 5 на рис. 2.26 и изображением 5 на рис. 2.27 (реконструкция с предварительной фильтрацией шума) показывает, что путь б) (предварительная фильтрация шума, а затем — реконструкция) дает меньшую погрешность реконструкции  $\sigma_{\text{rel}}(\alpha_{\text{opt}})$ , чем путь а) (реконструкция без предварительной фильтрации шума). К еще более четкому результату мы приходим в случае импульсного шума (ср. кривую 4 на рис. 2.26 и изображение 4 на рис. 2.27 с кривой 6 на рис. 2.26 и изображением 6 на рис. 2.27). Это говорит о том, что метод конечных сумм с регуляризацией Тихонова (вариант 4) достаточно эффективно устраняет смазывание, но менее эффективно подавляет шумы. Поэтому целесообразно перед обращением к методу регуляризации Тихонова выполнить фильтрацию шумов такими специальными методами, как адаптивная фильтрация Винера, медианная фильтрация, ранговая фильтрация и т.п.

Теперь рассмотрим для сравнения, как влияют зашумленность смазанного изображения и погрешность ФРТ на результат реконструкции изображения, если смазывание (прямая задача) выполняется с использованием "граничных условий" (по функциям `fspecial` и `imfilter`), а реконструкция (обратная задача) выполняется параметрической фильтрацией Винера (по функции `deconvwnr`). Для этого были записаны следующие команды (см. `text3Werr.m`).

```
Delta=20; %Точное значение Delta
%Точная функция рассеяния точки (ФРТ):
PSF=fspecial('motion',Delta,0);
mot=imfilter(wp,PSF,'symmetric'); %смазанное изображение
%Добавление гауссова 1%-ного шума к mot:
randn('state',0); %установка генератора случ. чисел в нач. состоянии
motgauss1=normnoise(mot,1);
%Добавление гауссова 3%-ного шума к mot:
motgauss3=normnoise(mot,3);
%Добавление импульсного шума:
motimp=double(imnoise(uint8(mot),'salt & pepper',0.03));
%Фильтрация гауссова 3%-ного шума адаптивной фильтрацией Винера:
Wiengauss3=wiener2(motgauss3,[5 5]);
%Фильтрация импульсного шума медианной фильтрацией:
medimp=medfilt2(motimp,[5 5]);
```

На рис. 2.28 (см. `text3Werr.m`) представлены полученные изображения `mot`, `motga-`

uss1, motgauss3, motimp, Wiengauss3 и medimp.

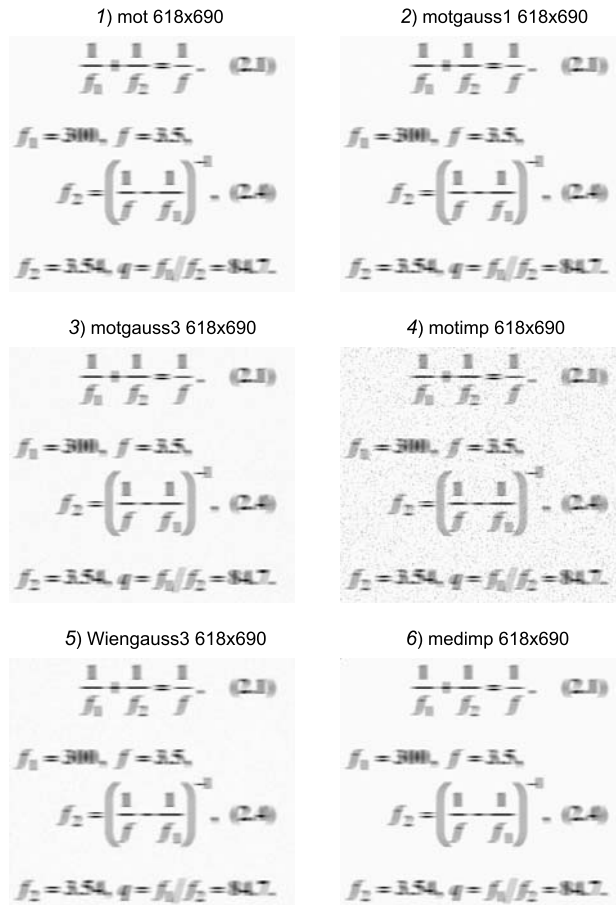


Рис. 2.28. Изображения со смазыванием, зашумленностью и фильтрацией

Затем в цикле по  $K$  (2.46) были рассчитаны относительные погрешности реконструкции  $\sigma_{\text{rel}}(K)$  методом параметрической фильтрации Винера, представленные на рис. 2.29 (см. text3Werr.m).

Для расчета кривых на рис. 2.29 были выполнены следующие операции:

1 — реконструкция смазанного изображения mot без шума, но с ошибочной ФРТ:

```
PSF1=fspecial('motion',Delta+2,0); %ошибочная ФРТ
```

```
w_Kperr=deconvwnr(uint8(mot),PSF1,Ki);
```

```
sig_relKperr(i)=rmsd(w_Kperr,wp);
```

2 — реконструкция смазанного и зашумленного 1%-ным гауссовым шумом изображения motgauss1:

```
w_Kpgauss1=deconvwnr(uint8(motgauss1),PSF,Ki);
```

```
sig_relKpgauss1(i)=rmsd(w_Kpgauss1,wp);
```

3 — реконструкция смазанного и зашумленного 3%-ным гауссовым шумом изображения motgauss3:

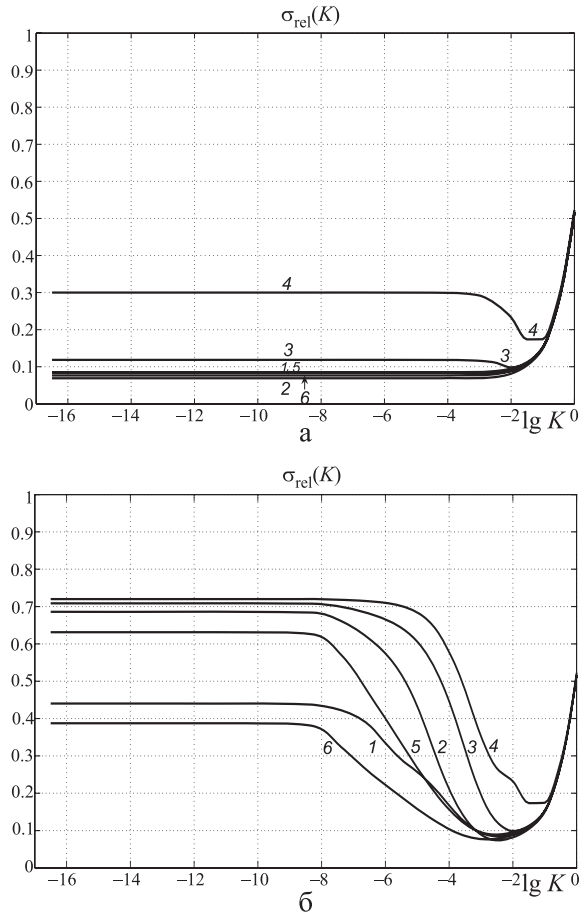


Рис. 2.29. Зависимости относительной погрешности  $\sigma_{rel}(K)$  реконструкции смазанных и зашумленных изображений методом параметрической фильтрации Винера  
 а — MatLab 7.0.1      б — MatLab 7.10.0

```
w_Kpgauss3=deconvwnr(uint8(motgauss3),PSF,Ki);
```

```
sig_relKpgauss3(i)=rmsd(w_Kpgauss3,wp);
```

4 — реконструкция смазанного и зашумленного импульсным шумом изображения `mot-imp`:

```
w_Kpimp=deconvwnr(uint8(motimp),PSF,Ki);
```

```
sig_relKpimp(i)=rmsd(w_Kpimp,wp);
```

5 — реконструкция смазанного изображения, у которого предварительно отфильтрован 3%-ный гауссовый шум адаптивной фильтрацией Винера:

```
w_KpWgauss3=deconvwnr(uint8(Wiengauss3),PSF,Ki);
```

```
sig_relKpWgauss3(i)=rmsd(w_KpWgauss3,wp);
```

6 — реконструкция смазанного изображения, у которого предварительно отфильтрован импульсный шум медианной фильтрацией:

```
w_Kpmimp=deconvwnr(uint8(medimp),PSF,Ki);
sig_relKpmimp(i)=rmsd(w_Kpmimp,wp);
```

При этом на рис. 2.29а представлены результаты расчета погрешности  $\sigma_{\text{rel}}(K)$  в результате использования функции `deconvwnr` из MatLab 7.0.1, а на рис. 2.29б — `deconvwnr` из MatLab 7.10.0, причем в обоих случаях обращение к `deconvwnr` выполнено в виде: `deconvwnr(uint8(mot),PSF1,Ki)`. Если же обращение к `deconvwnr` выполнить в виде: `deconvwnr(mot,PSF1,Ki)`, то кривые  $\sigma_{\text{rel}}(K)$  (непредставленные на рисунке) пойдут значительно выше.

На рис. 2.30 (см. `text4Werr.m`) приведены изображения, реконструированные методом параметрической фильтрации Винера при  $K = K_{\text{opt}}$ , причем значения  $K_{\text{opt}}$  получены по кривым 1–6 на рис. 2.29 согласно (2.43).

На рис. 2.30:

```
1 — w_Kperr=deconvwnr(uint8(mot),PSF1,1e-4);
   %без шума, но с ошибочной ФРТ
2 — w_Kpgauss1=deconvwnr(uint8(motgauss1),PSF,1e-4);
   %с 1%-м гауссовым шумом
3 — w_Kpgauss3=deconvwnr(uint8(motgauss3),PSF,1e-2);
   %с 3%-м гауссовым шумом
4 — w_Kpimp=deconvwnr(uint8(motimp),PSF,10^(-1.2));
   %с импульсным шумом
5 — w_Kpwgauss3=deconvwnr(uint8(Wiengauss3),PSF,1e-4);
   %с предварительной фильтрацией 3%-ного гауссова шума
   %адаптивным фильтром Винера
6 — w_Kpmimp=deconvwnr(uint8(medimp),PSF,1e-4);
   %с предварительной фильтрацией импульсного шума
   %медианным фильтром
```

Анализ рис. 2.28–2.30 и сравнение их с рис. 2.25–2.27 позволяет сделать следующие выводы (дополняющие выводы 1)–7), сделанные выше на стр. 113, 116 и 122).

8) Если в отсутствие шумов и погрешности ФРТ и при наличии только смазывания метод параметрической фильтрации Винера дает заметно большую погрешность реконструкции  $\sigma_{\text{rel}}$ , чем вариант 4 (см. вывод 5) на стр. 116), то при наличии не только смазывания, но и шумов и погрешности в ФРТ значения  $\sigma_{\text{rel}}(K_{\text{opt}})$  методом Винера несколько меньше значений  $\sigma_{\text{rel}}(\alpha_{\text{opt}})$  4-м вариантом (ср. рис. 2.29, 2.30 с рис. 2.26, 2.27). Это говорит о том, что метод фильтрации Винера несколько более устойчив, чем метод регуляризации Тихонова по отношению к шумам на изображении и погрешностям знания ФРТ (в данном случае погрешности знания величины смаза  $\Delta$ ).

**Портретное изображение.** Теперь наряду с текстовым изображением (см. стр. 101) рассмотрим обработку портретного изображения, в первую очередь, изображения человека. Специфика портретного изображения состоит в том, что, во-первых, это обычно непрерывное изображение, без пустот (в отличие от текста), во-вторых, на нем, как правило, нет яркой точки, по которой можно оценить параметры смаза, и в-третьих, при оценке качества реконструкции портретного изображения следует

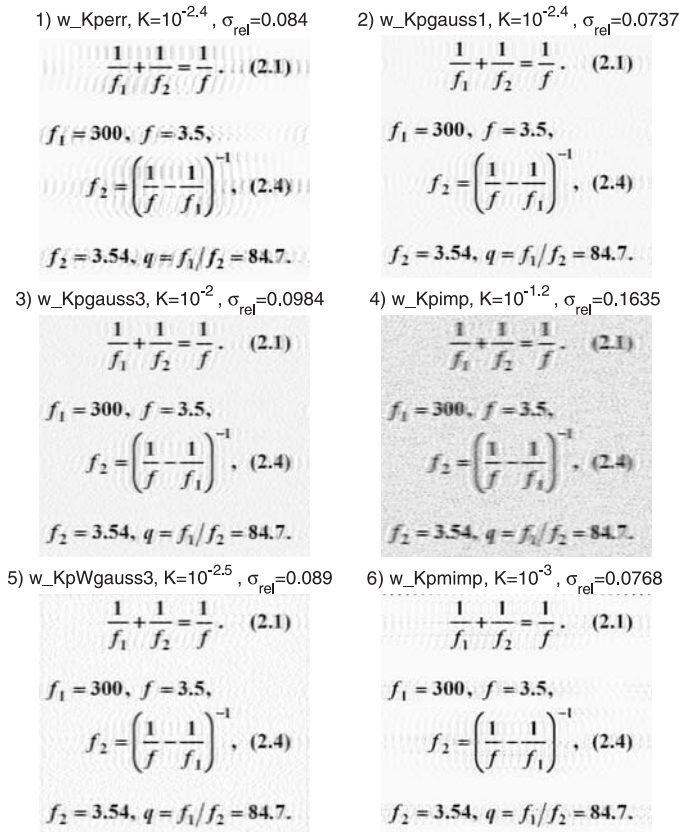


Рис. 2.30. Реконструкция смазанных изображений с погрешностями в ФРТ и с шумами методом параметрической фильтрации Винера при  $K = K_{\text{opt}}$ .  
Размер каждого изображения  $618 \times 690$

использовать не только такой "технический показатель", как  $\sigma_{\text{rel}}$  (см. (2.41), (2.44)), но и визуальное эстетическое восприятие.

**Изображение со смазыванием и гауссовым шумом.** В качестве портретного изображения рассмотрим серое изображение `cameraman.tif` размером  $m = 256$  строк,  $n = 256$  столбцов. Данное изображение неоднократно использовалось в ряде публикаций ([35, 46, 105, 112, 148, 150] и др.) в качестве модельного изображения (фантома). Это изображение в работе [35, с. 111] названо изображением со средним уровнем детализации.

Выполнены следующие команды (*прямая задача*) (см. `cam1.m`):

```
w=imread('cameraman.tif'); %256x256 uint8
w=double(w); %256x256 double
figure, imshow(uint8(w)), title('w 256x256')
Delta=10; %длина смаза в пкс (double)
%Смазанное изображение с усечением:
```

```

g1=smearing(w,Delta,0,'truncation'); %256x246 double
%Dобавление 1%-ного гауссова шума к g1:
randn('state',0); %установка генератора случ. чисел в нач. состоянии
per=1; %процент шума
g1n=normnoise(g1,per); %256x246 double
%Смазанное изображение с размытыми краями:
g2=smearing(w,Delta,0,'diffusion'); %256x266 double
%Dобавление 1%-ного гауссова шума к g2:
g2n=normnoise(g2,per); %256x266 double
figure, imshow(uint8(g2n)), title('g2n 256x266')
    
```

На рис. 2.31 (см. `cam1.m`) представлено исходное (точное) изображение  $w$  шириной  $n = 256$ , а также смазанные ( $\Delta = 10$ ) и зашумленные гауссовым шумом  $\|\delta g\|/\|g\| = 0.01 = 1\%$  изображения с усечением  $\hat{g} = g1n$  шириной  $n - \Delta = 246$  и с размытыми краями  $g = g2n$  шириной  $n + \Delta = 266$ .

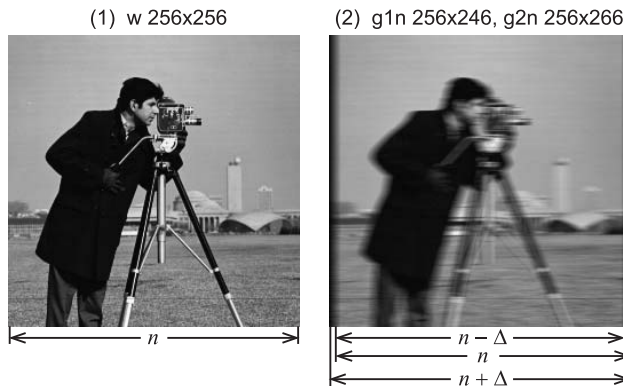


Рис. 2.31. (1) — исходное изображение  $w$ ,  
 (2) — смазанные и зашумленные изображения  $\hat{g}$  и  $g$   
 ( $\Delta = 10$ , гауссовый шум,  $\|\delta g\|/\|g\| = 1\%$ )

Далее решалась *обратная задача*. На рис. 2.32 (см. `cam1.m`) приведены кривые зависимости относительной погрешности реконструкции  $\sigma_{\text{rel}}(\alpha)$  при 1%-ном шуме и при  $p = 1$  (см. (2.29)) для вариантов 1–5.<sup>5</sup> Используются собственные  $m$ -функции `desmearingf` (для вариантов 1, 2), `desmearingq` (для вариантов 3–5) и `rmsd`. При этом параметр регуляризации  $\alpha$  изменялся по закону типа (2.45).

На рис. 2.32 для сравнения приведены также кривые  $\sigma_{\text{rel}}(K)$  для метода параметрической фильтрации Винера (варианты `6c` и `6s`). В вариантах `6c` и `6s` в прямой задаче использованы внешние  $m$ -функции `fspecial` и `imfilter` с опциями `'circular'` (соответствует "граничному условию" `periodic`) и `'symmetric'` (соответствует `reflecti-`

<sup>5</sup>Напомним, что в вариантах 1, 3 и 5 используется изображение `g1n` (с усечением), а в вариантах 2 и 4 — изображение `g2n` (с размытием краев).

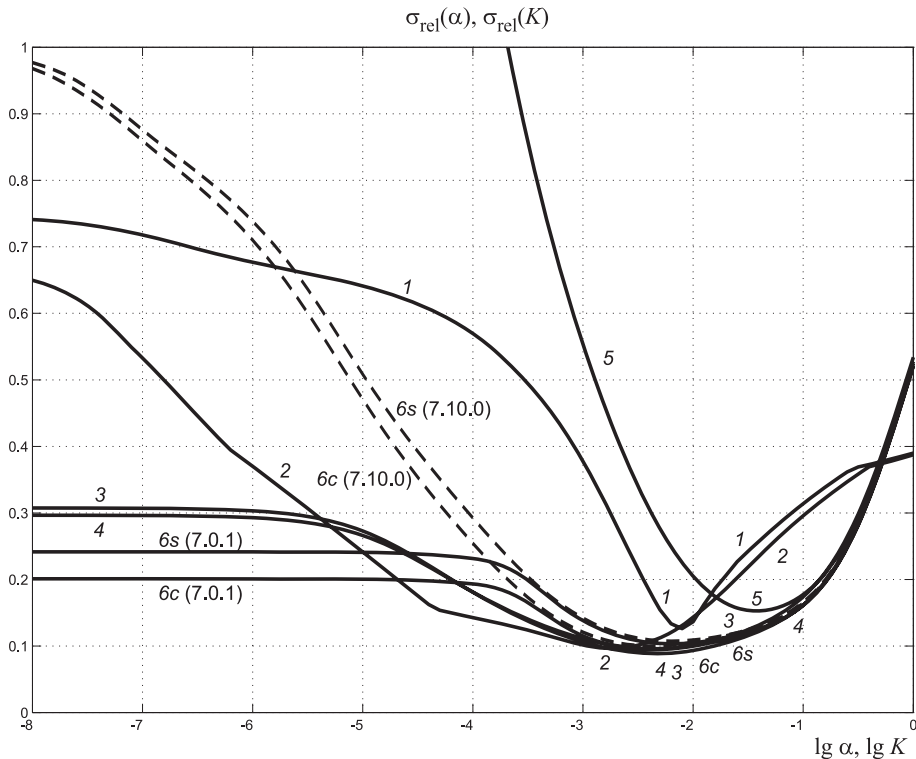


Рис. 2.32. Зависимости  $\sigma_{\text{rel}}(\alpha)$  для вариантов 1–5 и  $\sigma_{\text{rel}}(K)$  для вариантов 6с и 6s; смаз  $\Delta = 10$ , относительный уровень гауссова шума  $\|\delta g\|/\|g\| = 1\%$ ; m-функция `wiener2` не использована

ve). В обратной задаче использована внешняя m-функция `deconvwnr`, причем параметр  $K$  изменялся по закону типа (2.46).

Заметим, что на рис. 2.32 приведены кривые  $\sigma_{\text{rel}}(K)$ , полученные с использованием функции `deconvwnr` из подверсий MatLab 7.0.1 и MatLab 7.10.0, причем обращение к `deconvwnr` выполнено в виде (см. `cam1.m`): `Wnr_c=deconvwnr(uint8(cn),...)` и `Wnr_s=deconvwnr(uint8(sn),...)`. Если же обращаться: `Wnr_c=deconvwnr(cn,...)` и `Wnr_s=deconvwnr(sn,...)`, то (неприведенные) кривые  $\sigma_{\text{rel}}(K)$  при  $\lg K \lesssim -5$  пойдут заметно выше, а при  $\lg K \gtrsim -4$  будут практически совпадать (отметим, что на рис. 2.13 и 2.16 эффект был противоположный: пунктирные кривые шли выше штрихпунктирных). При этом при расчете зависимостей  $\sigma_{\text{rel}}(\alpha)$  и  $\sigma_{\text{rel}}(K)$  не использовались для фильтрации шума такие методы, как адаптивная фильтрация Винера (внешняя m-функция `wiener2`).

На рис. 2.33 (см. `cam2.m`) представлены результаты реконструкции изображения, выполненные согласно вариантам 1–4 при  $\alpha = \alpha_{\text{opt}}$  (см. (2.42)) и вариантам 6с и 6s при  $K = K_{\text{opt}}$  (см. (2.43)) при уровне гауссова шума  $\|\delta g\|/\|g\| = 1\%$ .

Для построения изображений на рис. 2.33 были использованы следующие опера-





Рис. 2.33. Реконструкция смазанного и зашумленного 1%-ным гауссовым шумом изображения без использования  $m$ -функции `wiener2` при  $\alpha = \alpha_{\text{opt}}$  (варианты 1–4) и  $K = K_{\text{opt}}$  (варианты 6с и 6s)

торы ( $[m, n] = \text{size}(w)$ ):

```
(1) w_alpha1=desmearingf(g1n,Delta,0,'var1',10^(-2.1),1,'mod',m,n);
(2) w_alpha2=desmearingf(g2n,Delta,0,'var2',10^(-2.75),1,'mod',m,n);
(3) w_alpha3=desmearingq(g1n,Delta,0,'var3',10^(-2.3),'mod',m,n);
(4) w_alpha4=desmearingq(g2n,Delta,0,'var4',10^(-2.3),'mod',m,n);
(6c) PSF=fspecial('motion',Delta,0); c=imfilter(w,PSF,'circular');
      cn=normnoise(c,1); Wnr_c=deconvwnr(uint8(cn),PSF,10^(-2,4));
(6s) s=imfilter(w,PSF,'symmetric'); sn=normnoise(s,1);
      Wnr_s=deconvwnr(uint8(sn),PSF,10^(-2.2));
```

Видно, что все варианты отягощены артефактами, или ложными волнами на изображении — проявление эффекта Гиббса. При этом наименьший эффект Гиббса и наилучшее восстановление дает вариант 4 как с позиций зрительного восприятия изображения, так и по критерию минимума СКО  $\sigma_{\text{rel}}$ .

На рис. 2.34 (см. `cam3.m`) приведены погрешности реконструкции  $\sigma_{\text{rel}}(\alpha)$  для вариантов 1–5 и  $\sigma_{\text{rel}}(K)$  для вариантов 6с и 6s при 1%-ном гауссовом шуме и при использовании m-функции `wiener2` с окном [3 3] для фильтрации шума (перед обращением к функциям `desmearingf`, `desmearingq` и `deconvwnr`). Видим, что ход кривых по сравнению с рис. 2.32 практически не изменился, т.е. фильтрация гауссова шума адаптивным фильтром Винера (по m-функции `wiener2`) недостаточно эффективна.

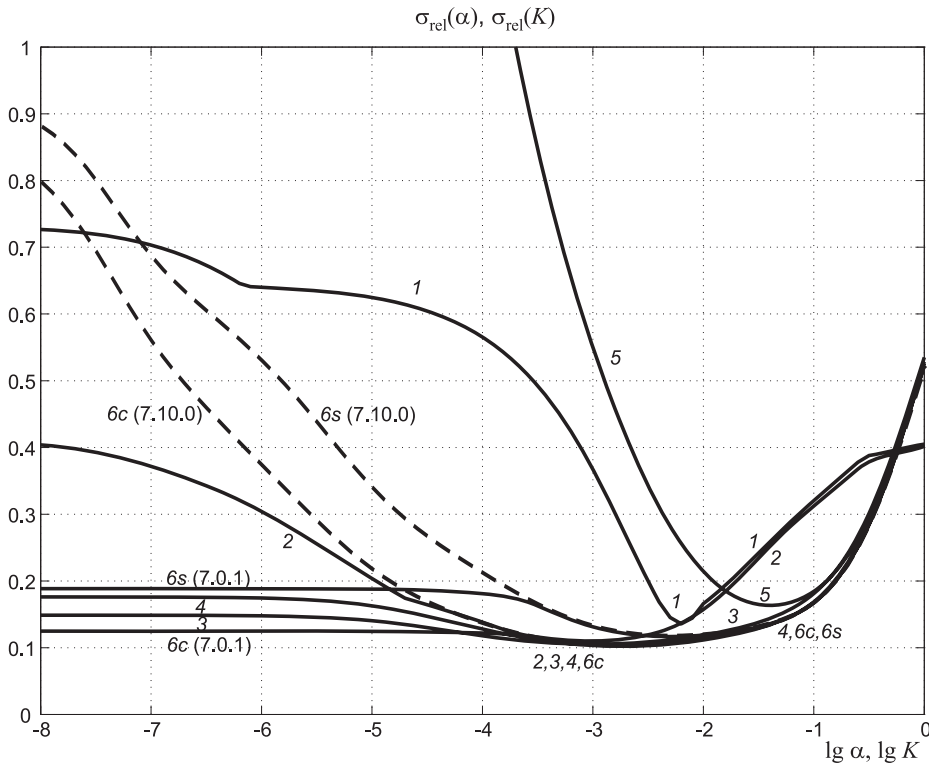


Рис. 2.34. Зависимости  $\sigma_{\text{rel}}(\alpha)$  для вариантов 1–5 и  $\sigma_{\text{rel}}(K)$  для вариантов 6с и 6s; смаз  $\Delta = 10$ , относительный уровень гауссова шума  $\|\delta g\|/\|g\| = 1\%$ ; использована m-функция `wiener2`

**Изображение со смазыванием и импульсным шумом.** Теперь рассмотрим случай, когда смазанное изображение зашумлено импульсным (двоичным) шумом (типа 'salt & pepper'). Вместо команд добавления 1%-ного гауссова шума к `g1` и к `g2` были реализованы следующие команды (*прямая задача*):

```
%Добавление импульсного шума к g1:
g1n=double(imnoise(uint8(g1),'salt & pepper',0.03));
%Добавление импульсного шума к g2:
g2n=double(imnoise(uint8(g2),'salt & pepper',0.03));
```

На рис. 2.35 (см. `cam4.m`) представлено исходное (точное) изображение `w` шириной  $n = 256$ , а также смазанные ( $\Delta = 10$ ) и зашумленные импульсным шумом изобра-

жения с усечением  $g1n$  шириной  $n - \Delta = 246$  и с размытыми краями  $g2n$  шириной  $n + \Delta = 266$ .

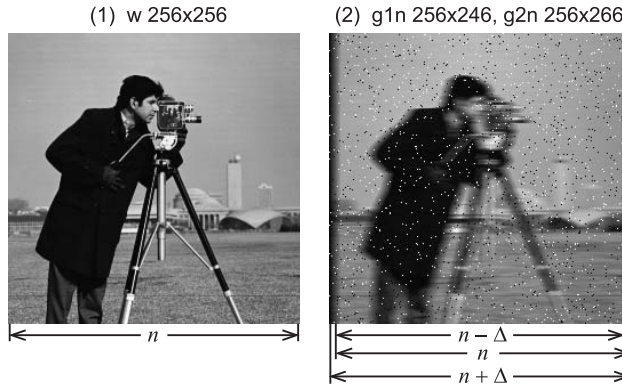


Рис. 2.35. (1) — исходное изображение, (2) — смазанные и зашумленные изображения ( $\Delta = 10$ , импульсный шум,  $d=0.03$ )

Далее решалась *обратная задача*.

На рис. 2.36 (см. `cam4.m`) приведены относительные погрешности реконструкции  $\sigma_{rel}(\alpha)$  для вариантов 1–5 и  $\sigma_{rel}(K)$  для вариантов 6с и 6s при импульсном шуме, причем для фильтрации шума не использовалась медианная фильтрация (внешняя m-функция `medfilt2`). Видим, что без дополнительной фильтрации импульсного шума методы регуляризации Тихонова и параметрической фильтрации Винера реконструируют изображение со значительной погрешностью ( $\sigma_{rel} > 0.2$ ).

На рис. 2.37 (см. `cam5.m`) приведены погрешности  $\sigma_{rel}(\alpha)$  для вариантов 1–5 и  $\sigma_{rel}(K)$  для вариантов 6с и 6s при импульсном шуме и при предварительном использовании m-функции `medfilt2` с окном `[3 3]` для фильтрации шума (перед обращением к функциям `desmearingf`, `desmearingq` и `deconvwnr`). Видим, что медианная фильтрация импульсного шума привела к заметному снижению погрешности  $\sigma_{rel}$ .

На рис. 2.38 (см. `cam6.m`) представлены результаты реконструкции изображения, выполненные согласно вариантам 1–4 при  $\alpha = \alpha_{opt}$  и варианты 6с и 6s при  $K = K_{opt}$  в случае импульсного шума при использовании m-функции `medfilt2`.

Далее дана таблица, суммирующая результаты для портретного изображения.

В таблице приведены значения  $\alpha_{opt}$ ,  $\sigma_{rel}(\alpha_{opt})$ ,  $K_{opt}$  и  $\sigma_{rel}(K_{opt})$ , полученные в результате обработки изображения `cameraman.tif` для вариантов 1–5, 6с и 6s при  $\Delta = 10$ ,  $p = 1$ , 1%-ном гауссовом шуме и импульсном шуме ( $d = 0.03$ ) без использования адаптивной фильтрации Винера (`wiener2`) и медианной фильтрации (`medfilt2`), а также с их использованием. В таблице приведены также значения  $\alpha_{opt}$ ,  $\sigma_{rel}(\alpha_{opt})$ ,  $K_{opt}$ ,  $\sigma_{rel}(K_{opt})$  при других значениях гауссова шума:  $\|\delta g\|/\|g\| = 0, 0.001, 0.03$  и  $0.09$  (без использования `wiener2`).

Анализ рис. 2.31–2.38 и, особенно, таблицы дает возможность сделать следующие выводы (ср. с выводами 1)–8) на стр. 113, 116, 122 и 125, сделанными применительно к обработке текстового изображения):

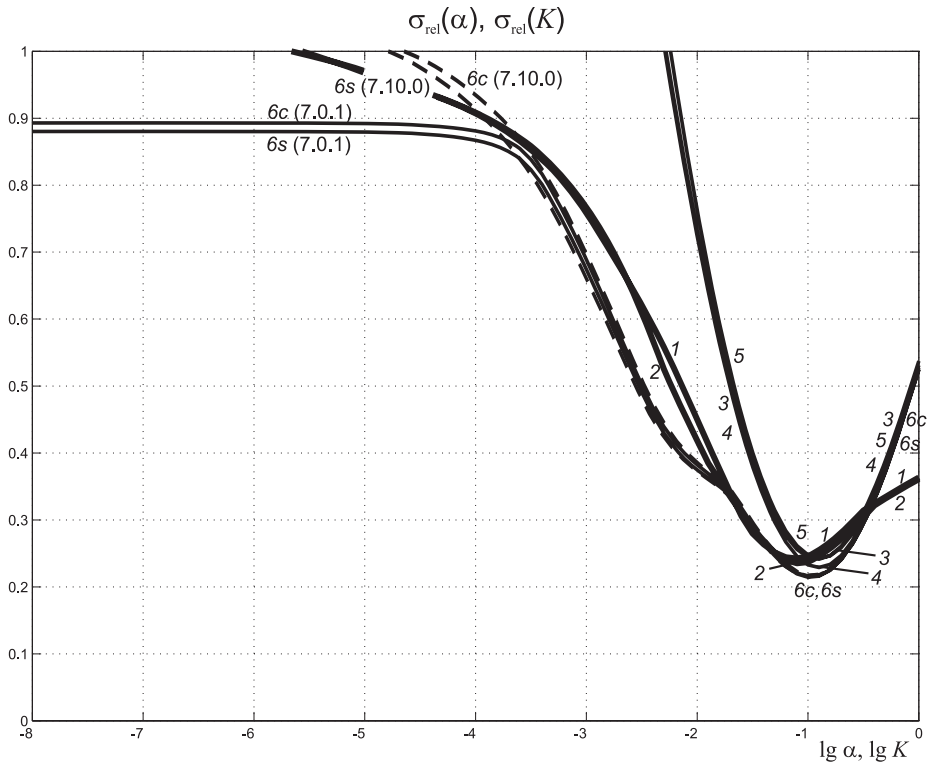


Рис. 2.36. Зависимости  $\sigma_{rel}(\alpha)$  для вариантов 1–5 и  $\sigma_{rel}(K)$  для вариантов 6с и 6s; смаз  $\Delta = 10$ , импульсный шум: доля зашумленных пикселей, или плотность шума  $d=0.03$ ; m-функция `medfilt2` не использована

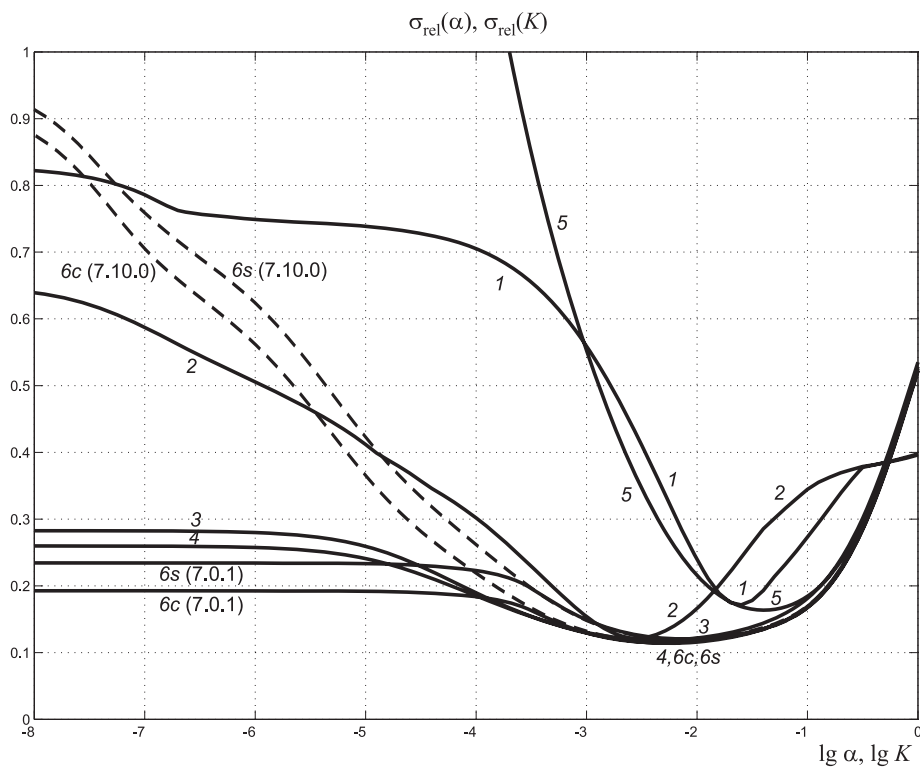


Рис. 2.37. Зависимости  $\sigma_{rel}(\alpha)$  для вариантов 1–5 и  $\sigma_{rel}(K)$  для вариантов 6c и 6s; смаз  $\Delta = 10$ , импульсный шум:  $d=0.03$ ; использована m-функция medfilt2



Рис. 2.38. Реконструкция смазанного и зашумленного импульсным шумом изображения с использованием  $m$ -функции `medfilt2` при  $\alpha = \alpha_{\text{opt}}$  (варианты 1–4) и  $K = K_{\text{opt}}$  (варианты 6с и 6s)

Таблица

Отн. шум $\ \delta g\ /\ g\ $	m-функция и вариант							
	desmearingf.m ( $p = 1$ )				desmearingq.m			
	1		2		3		4	
	$\lg \alpha_{opt}$	$\sigma_{rel}(\alpha_{opt})$	$\lg \alpha_{opt}$	$\sigma_{rel}(\alpha_{opt})$	$\lg \alpha_{opt}$	$\sigma_{rel}(\alpha_{opt})$	$\lg \alpha_{opt}$	$\sigma_{rel}(\alpha_{opt})$
0	-2.2	0.128	< -7	0.008	< -5	0.038	< -7	< 5e-5
гаус. шум, 0.001=0.1%	-2.1	0.124	-5.3	0.029	-4.2	0.044	-4.3	0.024
гаус. шум, 0.01=1%	-2.1	0.126	-2.7	0.096	-2.3	0.096	-2.3	0.088
гаус. шум, 0.03=3%	-2.1	0.139	-2.3	0.126	-1.6	0.135	-1.6	0.127
гаус. шум, 0.09=9%	-1.8	0.191	-1.8	0.182	-1.2	0.184	-1.1	0.174
гаус. шум, 0.01=1% с wiener2.m	-2.2	0.136	-3.0	0.110	-3.1	0.105	-2.7	0.102
имп. шум без medfilt2.m	-1.1	0.243	-1.1	0.236	-0.9	0.241	-0.9	0.229
имп. шум с medfilt2.m	-1.6	0.171	-2.5	0.121	-2.3	0.119	-2.2	0.114

Отн. шум $\ \delta g\ /\ g\ $	m-функция и вариант					
	desmearingq.m			deconvwnr.m (7.0.1 и 7.10.0)		
	5		6с		6s	
	$\lg \alpha_{opt}$	$\sigma_{rel}(\alpha_{opt})$	$\lg K_{opt}$	$\sigma_{rel}(K_{opt})$	$\lg K_{opt}$	$\sigma_{rel}(K_{opt})$
0	-1.4	0.152	-3.6	0.056	-2.5	0.092
гаус. шум, 0.001=0.1%	-1.5	0.152	-3.4	0.059	-2.5	0.092
гаус. шум, 0.01=1%	-1.4	0.153	-2.4	0.094	-2.2	0.103
гаус. шум, 0.03=3%	-1.4	0.160	-1.7	0.125	-1.7	0.126
гаус. шум, 0.09=9%	-1.1	0.191	-1.2	0.167	-1.2	0.167
гаус. шум, 0.01=1% с wiener2.m	-1.4	0.163	-2.8	0.104	-2.3	0.116
имп. шум без medfilt2.m	-0.9	0.243	-1.0	0.216	-1.0	0.215
имп. шум с medfilt2.m	-1.4	0.163	-2.3	0.115	-2.1	0.120

а) В отсутствие шумов ( $\|\delta g\|/\|g\| = 0$ ) и при наличии только смазывания, как следует из таблицы, метод ПФ с регуляризацией Тихонова (функция `desmearingf.m`, варианты 1, 2) дает значительно бóльшую погрешность  $\sigma_{rel}(\alpha_{opt})$ , чем метод конечных сумм (квадратур) с регуляризацией Тихонова (функция `desmearingq.m`, варианты 3, 4). Метод параметрической фильтрации Винера, использующий "граничные условия" (функция `deconvwnr.m`, варианты 6с, 6s), дает также бóльшую погрешность  $\sigma_{rel}(K_{opt})$ , чем варианты 3 и 4. Этот вывод подтверждает выводы 2) на стр. 113 и 5) на стр. 116. Для иллюстрации сказанного на рис. 2.39 (см. `cam7.m`) приведены изображения, реконструированные вариантами 3, 4, 6с и 6s. Видно, что вариант 4 дал практически точное восстановление изображения.

б) При  $\|\delta g\|/\|g\| = 0$  метод, использующий размытие краев изображения (варианты 2 и 4), более точен, чем метод, использующий усечение изображения (варианты 1 и 3). Это подтверждает вывод 1) на стр. 113.

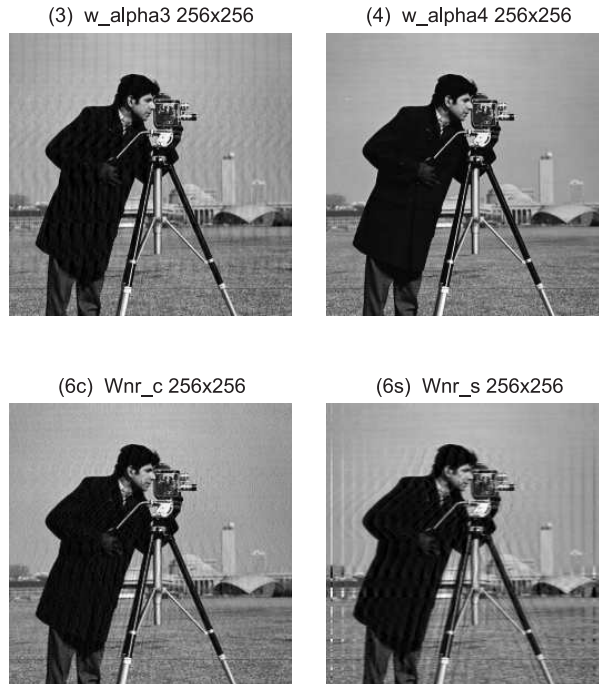


Рис. 2.39. Реконструкция смазанного (без шума) изображения при  $\alpha = \alpha_{opt}$  (варианты 3 и 4) и  $K = K_{opt}$  (варианты 6c и 6s)

в) Если на смазанное изображение наложен 1%-ный гауссовый шум, то наименьшую погрешность  $\sigma_{rel}$  восстановления изображения имеет вариант 4 как без использования функции `wiener2`, так и с использованием `wiener2`. При этом использование функции `wiener2` не понижает погрешность  $\sigma_{rel}$ , как при обработке текстового изображения, а повышает  $\sigma_{rel}$ .

г) Если же на смазанное изображение наложен импульсный шум, то реконструкция изображения без использования функции `medfilt2` дает значительную погрешность  $\sigma_{rel}$ , но использование функции `medfilt2` понижает заметно (вдвое) погрешность  $\sigma_{rel}$ . Это говорит о том, что метод регуляризации Тихонова и метод параметрической фильтрации Винера достаточно эффективно самостоятельно фильтруют гауссовый шум на портретном изображении, но значительно хуже фильтруют импульсный шум и требуется применение специального метода (медианной фильтрации и т.п.).

д) Если гауссовый шум составляет  $\gtrsim 3\%$ , то вариант 4, а также варианты 6c и 6s дают примерно одинаковую погрешность  $\sigma_{rel}$ . Это говорит о том, что когда главным фактором в искажении изображения является шум, а не смазывание, методы регуляризации Тихонова и параметрической фильтрации Винера становятся сопоставимыми по точности реконструкции.

е) Наименее точным является вариант 5 (как и в случае текстового изображения, см. стр. 116), что можно объяснить несоответствием прямой задачи (2.18) и обратной задачи (2.36), а именно тем, что в прямой задаче используются значения  $w$  правого



края изображения шириной  $\Delta$  (см. рис. 2.8), а в обратной задаче они исключаются из рассмотрения, т.е. полагаются равными нулю.

Обработка других изображений: `girl.tif` из работ [27], [35, с. 471], `plane.bmp` из работы [168, р. 196] и др. подтверждает сделанные выводы.

**Сравнение с результатами других публикаций.** Результаты представленные в работе [150, table 2], показывают, что при искажении изображения `cameraman.tif` гауссовым окном  $30 \times 30$  пкс (что примерно соответствует  $\Delta = 10$ , если судить по функции рассеяния точки PSF [150, Figure 2]) и зашумлении 1%-ным шумом ( $\text{SNR} = 40$ ) реконструкция изображения алгоритмами `deblurring` и `re-blurring` (с CG-регуляризацией [151]) выполняется с относительной погрешностью примерно 0.23 для "граничного условия" `periodic`, 0.20 для `reflective` и 0.19 для `anti-reflective`, т.е. заметно грубее, чем вариантом 4 (а также 1–3, 6с и 6s). Результаты, аналогичные [150], даны также в работе [148, table 7.2]. Правда, в работах [148, 150] PSF соответствует скорее дефокусированию, чем смазыванию изображения, однако это непринципиально.

**Замечания.** Необходимо сделать следующие замечания относительно внешней  $m$ -функции `imshow` (а также `deconvwnr` и `imnoise`).

Например, на стр. 101 вывод изображения `wp` записан как `imshow(uint8(wp))`. Если же вывод записать как `imshow(wp)` или `imshow(im2uint8(wp))`, то качество вывода на дисплей и печати на принтер будет несколько хуже. Это можно объяснить тем, что `wp` имеет класс `double`, причем  $\min(wp(:))=0$ ,  $\max(wp(:))=250$ , а функция `imshow` особенно хорошо выводит изображение класса `uint8`, если  $\max \approx 250$ . Если же  $\max(wp(:))=1$ , то наилучший вывод будет при `imshow(wp)`, а при `imshow(uint8(wp))` выведется черное изображение. Это говорит о том, что в системе MatLab нередко матрицы класса `double` нормируются на интервал  $[0, 1]$ , а это может приводить к нежелательным последствиям. Сказанное относится и к функции `imnoise` [36, с. 156].

Необходимо сделать также замечание относительно функции `deconvwnr`, о чем уже говорилось выше. Например, на стр. 105 обращение к этой функции записано в виде: `Wnr=deconvwnr(uint8(mot),PSF,K)`; причем `PSF=fspecial('motion',20,0)`; где `mot` — класса `double`, равная `mot=imfilter(wp,PSF,'circular')`. В результате при  $K=1e-4$  СКО  $\sigma_{\text{rel}}$  равно  $\text{rmsd}(Wnr,wp)=0.0546$ . Если же обращение к `deconvwnr` записать в виде: `Wnr=deconvwnr(im2uint8(mot),PSF,K)` или `Wnr=deconvwnr(mot,PSF,K)`, то получим соответственно:  $\sigma_{\text{rel}} = 0.2112$  и  $0.3702$ , т.е. существенно бóльшие значения  $\sigma_{\text{rel}}$ . Это имеет место в подверсии MatLab 7.0.1, а, например, в подверсии MatLab 7.10.0 могут получиться заметно иные результаты, особенно, при малых  $K$  ( $\lesssim 10^{-5}$ ). Поэтому нужна повышенная аккуратность при обращении к функции `deconvwnr`.

**Быстрый алгоритм реконструкции смазанных изображений.** При обработке быстропротекающих процессов необходимо, чтобы реконструкция изображения выполнялась в реальном масштабе времени. Рассмотрим этот вопрос на примере быстро движущейся цели, например, самолета. Можно предложить следующий *быстрый алгоритм* реконструкции смазанного изображения цели на основе метода конечных сумм (квадратур) и регуляризации Тихонова (вариант 4) [108].

Решение (2.38) можно записать в виде:

$$w_{\alpha} = B_{\alpha}g, \quad (2.47)$$

где

$$B_\alpha = (\alpha E + A^T A)^{-1} A^T \quad (2.48)$$

— матрица  $n \times (n + \Delta)$ . Специфика решения (2.47) состоит в том, что матрица  $B_\alpha$  может быть рассчитана заранее и реконструкция изображения сведется к умножению "заготовленной" матрицы  $B_\alpha$  на каждый из  $m$  векторов-строк изображения  $g$ .

Оценим необходимое количество операций и время. Умножение матрицы  $B_\alpha$  размера  $n \times (n + \Delta)$  на вектор  $g$  длины  $(n + \Delta)$  можно записать в виде (ср. [102, с. 145]):

$$(w_\alpha)_i = \sum_{k=1}^{n+\Delta} (B_\alpha)_{ik} g_k, \quad i = 1, \dots, n. \quad (2.49)$$

На это потребуются  $n(n + \Delta)$  умножений (а также сложений, но мы их не будем учитывать, так как они выполняются на порядок быстрее, чем умножения). А на выполнение операций (2.49) для всех  $m$  строк изображения потребуются  $mn(n + \Delta)$  умножений. Например, при  $m = n = 400$  и  $\Delta \ll n$  потребуются  $\approx 64$  млн умножений. Если скорость компьютера  $\sim 1$  млрд оп/сек, то реконструкция изображения займет  $\sim 0.1$  сек.

Сделаем важное замечание. Матрица  $B_\alpha$  не зависит от номера строки  $j$  (при горизонтальном смазе), но зависит от величины смаза  $\Delta$  и параметра регуляризации  $\alpha$ , а также от размера матрицы носителя изображения  $M \times N$ . Поскольку размер  $M \times N$  обычно фиксирован, а величины  $\Delta$  и  $\alpha$  априори неизвестны, то следует заранее заготовить несколько матриц  $B_\alpha$  для ряда типичных значений  $\Delta$  и  $\alpha$ .

Если речь идет, например, о самолете — нарушителе границы, то служба первичного обнаружения цели должна оценить скорость, направление движения и высоту самолета и по этим данным служба обработки изображения может оценить  $\Delta$  (при известной экспозиции УРИ) и вызвать нужную матрицу  $B_\alpha$  с некоторым типичным значением  $\alpha$  (или две-три таких матрицы). И тогда за долю сек, т.е. в реальном масштабе времени будет выполнена реконструкция изображения самолета.

На рис. 2.40а (см. `FastRec.m`) — пример смазанного изображения самолета (`plane.bmp`,  $M = m = 510$ ,  $N = n + \Delta = 660$ ,  $\Delta = 20$ ). На изображение был наложен также 0.3%-ный гауссовый шум (малый шум был выбран потому, чтобы главным искажающим фактором было смазывание, а не зашумление). Из-за смаза на самолете не видны опознавательные знаки. Особенностью подобного смазанного изображения является то, что смазывание происходит не из-за движения фотоаппарата (в этом случае смазывается и самолет, и фон), а из-за движения самолета — в этом случае смазывается лишь часть изображения — самолет, а фон остается неподвижным. Это предъявляет повышенные требования к обработке изображения, а именно, в прямой задаче (смазывание моделируется) нужно смазывать, а в обратной задаче реконструировать лишь изображение самолета, не обрабатывая фон. Впрочем, если фон однороден, как в нашем примере, то можно обрабатывать целиком все изображение.

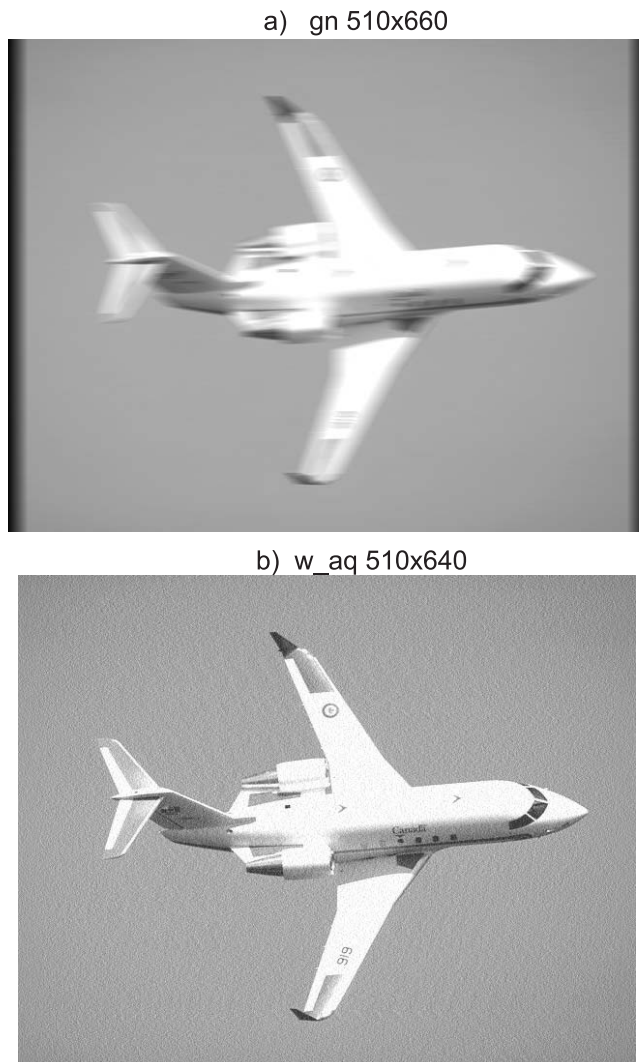


Рис. 2.40. Иллюстрация к быстрому алгоритму реконструкции смазанного изображения самолета: а — смазанное и зашумленное изображение с добавлением размытых краев, б — реконструированное изображение

На рис. 2.40б дано изображение самолета, реконструированное алгоритмом "заготовленной" матрицы согласно (2.47)–(2.49) ( $m \times n = 510 \times 640$ ,  $\alpha = 10^{-4}$ ,  $\sigma_{\text{rel}} = 0.0755$ ). На самолете стали видны опознавательные знаки. Реконструкция потребовала порядка 0.1 сек машинного времени.

Быстрый алгоритм можно также построить, если использовать метод преобразования Фурье с регуляризацией Тихонова [22, с. 267], [102, с. 184], однако, как мы выяснили выше, метод ПФ с регуляризацией менее точен, чем метод конечных сумм (квадратур) с регуляризацией, хотя требует меньше компьютерной памяти.

**Изображения, смазанные под углом, прямая и обратная задачи.** До сих пор мы рассматривали смазывание изображений под нулевым углом  $\theta$  (горизонталь-

ное смазывание). Теперь рассмотрим важный случай, когда угол  $\theta$ , вообще говоря, не равен нулю (более обще, не кратен  $180^\circ$ ) [41–43, 95, 106, 108].

**П р я м а я з а д а ч а.** Рассмотрим сначала прямую задачу — моделирование смазывания изображения. MatLab 7 включает в себя внешние m-функции `fspecial.m` и `imfilter.m`, с помощью которых можно выполнить, в частности, смазывание изображения под углом, например,

```
Delta=40; %величина смаза в пкс
theta=35; %угол смазывания в град
PSF=fspecial('motion',Delta,theta); %ФРТ
c=imfilter(w,PSF,'circular'); %смазывание (с опцией circular)
```

В `fspecial.m` заложен численный алгоритм смазывания под углом `theta`, в котором для вычисления разностной функции рассеяния точки (ФРТ)  $h(x, y)$  учитываются пиксели, отстоящие от каждой наклонной под углом  $\theta$  прямой линии не более, чем на 1 пкс. При этом используется билинейная интерполяция. В результате ФРТ получается в виде матрицы (вектора под углом). Далее в функции `imfilter.m` для получения смазанного изображения  $g(x, y)$  выполняется свертка истинного изображения  $w(x, y)$  с ФРТ  $h(x, y)$  (ср. (2.15)):

$$g(x, y) = \iint_D h(x - \xi, y - \eta) w(\xi, \eta) d\xi d\eta, \quad x, y \in D, \quad (2.50)$$

где  $D$  — область изображения (FOV), а непрерывное интегрирование реализуется в виде дискретного суммирования. На рис. 2.43с (см. `text9.m`) и 2.44с (см. `cam8.m`) (см. дальше) приведены примеры смазанных под углом изображений `c`, полученных с помощью функций `fspecial.m` и `imfilter.m`.

Основные особенности алгоритма, заложенного в функциях `fspecial` и `imfilter`: задача смазывания, как и задача дефокусирования (см. п. 2.5), решается как 2-мерная задача; используются так называемые "граничные условия" — ВСs<sup>6</sup>: `circular` (periodic), `symmetric` (reflective) и др.

Однако, как было сказано выше, целесообразнее рассматривать задачу чистого смазывания (без других искажений) как набор 1-мерных задач, а вместо внеграничных условий использовать *приемы усечения и размытия краев*. При этом вместо приема учета пикселей, близких к наклонной прямой линии, заложенного в функции `fspecial`, предлагаем использовать *прием (способ) поворота изображения*, который можно выполнить с помощью внешней функции `imrotate.m`.

На рис. 2.41 приведена схема 1 выполнения смазывания под углом  $\theta$  изображения  $w_{m \times n}$ , согласно которой выполняется поворот изображения  $w$  на угол  $\theta$  с помощью функции `imrotate` и делается переход к новой системе координат  $x, y$ , в которой ось  $x$  направлена вдоль смаза, а  $y$  — перпендикулярно ему. При этом при повороте с помощью функции `imrotate` область вне  $w$  (а также вне  $\hat{g}$ ) заполняется нулями.

<sup>6</sup>Как мы уже отметили, "граничные условия" целесообразнее называть внеграничными условиями — ОВСs.

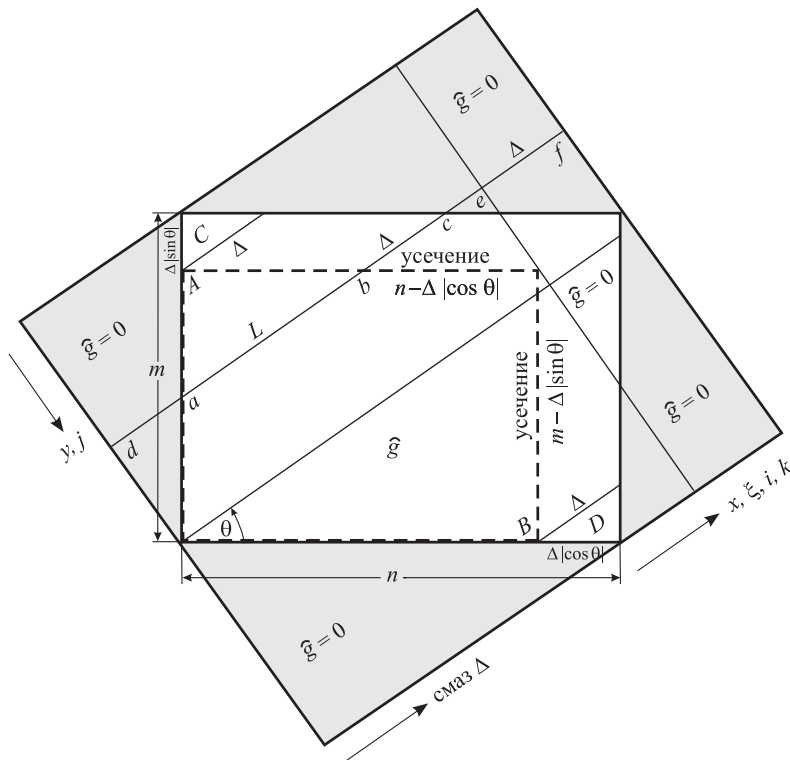


Рис. 2.41. Схема 1 смазывания изображения под углом с усечением и с использованием поворота

На рис. 2.41  $L$  — некоторая наклоненная под углом  $\theta$  прямая линия. Смысл точек  $d, a, b, c, e, f$  на линии  $L$  следующий. Значения  $w$  заданы на отрезке  $[a, c]$ , а после поворота — на отрезке  $[d, f]$ , а значения  $\hat{g}$  вычисляются на  $[a, b]$  по формуле типа (2.18). В результате, с учетом нулевых областей  $\hat{g}$ , значения  $\hat{g}$  будут получены на отрезке  $[d, e]$ . Добавим еще нулевую область  $[e, f]$  (для решения обратной задачи). В итоге значения  $\hat{g}$  будут получены на отрезке  $[d, f]$  (ср. с рис. 2.8), при этом ненулевые значения  $\hat{g}$  будут лишь на усеченном отрезке  $[a, b]$ .

**Обратная задача.** При решении обратной задачи (реконструкция  $w$ ) согласно схеме 1 (рис. 2.41) могут быть использованы варианты 1, 3 или 5, изложенные выше (см. стр. 89, 90 или 90). При этом ширина области поиска решения  $[a, b]$  в вариантах 1 и 5 или  $[a, c]$  в варианте 3 зависит от  $y$ , т.е. является переменной, что усложняет алгоритм как прямой, так и обратной задачи. Чтобы устранить это неудобство, можно при всех значениях  $y$  область задания  $\hat{g}$  сделать одной и той же, равной  $[d, f]$  (добавление нулевых значений  $\hat{g}$  не должно влиять на решение). Однако область поиска решения  $w$  нельзя делать равной  $[d, f] \forall y$ , так как решение  $w$  зависит от выбранной области  $[a, b]$ ,  $[a, c]$  или  $[d, f]$ .

Кроме того, схема 1 (рис. 2.41) при  $\theta \neq 0$  имеет еще следующие недостатки. Вари-

ант 3 при приближении  $L$  к точке  $A$  или  $B$  связан с решением сильно недоопределенной СЛАУ (область  $[a, c]$  значительно шире области  $[a, b]$ ), а варианты 1 и 5 вблизи  $A$  или  $B$  имеют очень сильное противоречие прямой и обратной задач. Наконец, в случае, когда  $L$  проходит через треугольную область  $C$  или  $D$ , в принципе невозможно восстановить  $w$ , так как нет для этого данных  $\hat{g}$  (область  $[a, b]$  вырождается в точку).

Все это говорит о том, что схема 1, представленная на рис. 2.41 является неэффективной при  $\theta \neq 0$ . Поэтому данную схему мы далее рассматривать будем лишь при  $\theta = 0$ .

Иначе обстоят дела, если использовать размытие краев у изображения  $g$  (ср. рис. 2.9).

**П р я м а я з а д а ч а.** На рис. 2.42 приведена схема 2 выполнения смазывания под углом  $\theta$  изображения  $w_{m \times n}$ , согласно которой выполняется поворот изображения  $w$  на угол  $\theta$  с помощью функции `imrotate` и делается переход к новым координатам  $x, y$  (ось  $x$  направлена вдоль смаза, а  $y$  — перпендикулярно ему). При этом функция `imrotate` заполняет нулями область вне  $w$ .

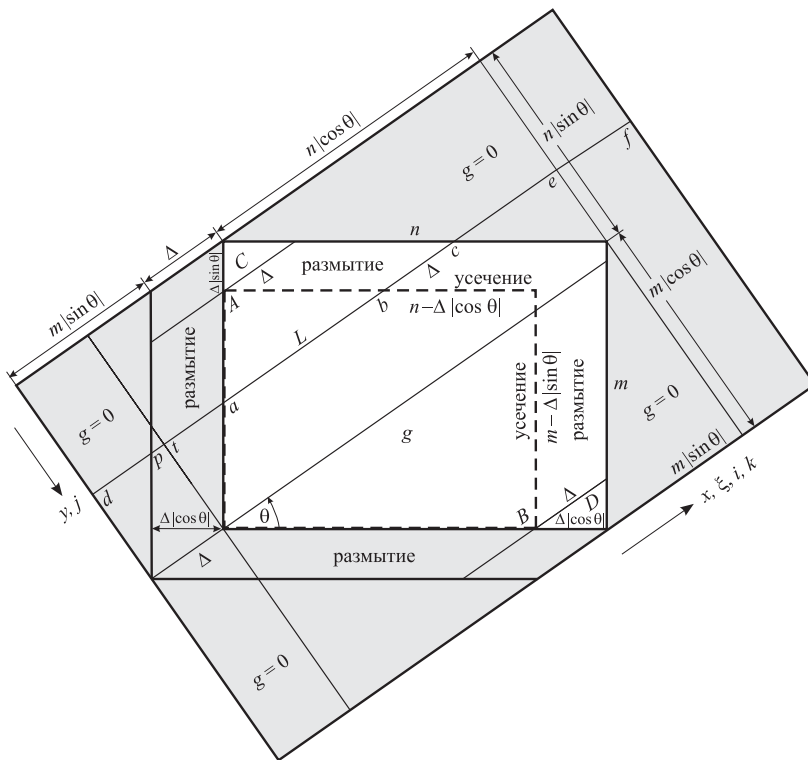


Рис. 2.42. Схема 2 смазывания изображения под углом с размытием краев и с использованием поворота

На рис. 2.42  $L$  — некоторая наклоненная под углом  $\theta$  прямая линия. Смысл точек  $d, p, t, a, b, c, e, f$  на линии  $L$  следующий. Значения  $w$  заданы на отрезке  $[a, c]$ , а после поворота — на отрезке  $[t, e]$ , значения  $\hat{g}$  с усечением вычисляются на  $[a, b]$  (как

в схеме 1 на рис. 2.41), а значения  $g$  с размытием краев — на  $[p, c]$  по формуле типа (2.20). При этом для получения массива  $u$  (см. (2.20)) делается дополнение повернутого массива  $w$  слева и справа нулями (с помощью функции `padarray`). В результате на отрезке  $[d, f]$  будут заданы значения  $u$ , по которым будут рассчитаны значения  $g$  с размытием краев на отрезке  $[d, e]$  по формуле типа (2.20).

Алгоритм получения смазанного под углом изображения  $g$  с размытыми краями (схема 2) реализован в рамках собственной  $m$ -функции `smearing.m`, о которой уже говорилось выше (на стр. 95). На рис. 2.43 и 2.44 в качестве примеров приведены изображения  $g$ , рассчитанные с помощью функции `smearing`.

**Обратная задача.** При решении обратной задачи (реконструкция  $w$ ) согласно схеме 2 (рис. 2.42) могут быть использованы варианты 2 или 4, изложенные выше (см. стр. 89 или 90). При этом ширина области поиска решения  $[p, c]$  в варианте 2 или  $[a, c]$  в варианте 4 зависит от  $y$ , т.е. является переменной, что усложняет алгоритм как прямой, так и обратной задачи. Чтобы устранить это неудобство, область задания  $g$  при всех значениях  $y$  целесообразно сделать одной и той же, равной  $[d, e]$  (добавление нулевых значений  $g$  не повлияет на решение). При этом решение  $w_\alpha$  ищем при всех  $y$  в одинаковой области, а именно, в  $[d, e]$  в варианте 2 и  $[t, e]$  в варианте 4. Принципиальных недостатков, присущих схеме 1, схема 2 не имеет.

После получения решения  $w_\alpha$  в системе координат  $x, y$  следует выполнить обратный поворот изображения  $w_\alpha$  (поэтому данную методику можно назвать *двойным поворотом изображения*) и удалить черную рамку вокруг  $w_\alpha$ .

Алгоритм реконструкции смазанного под углом изображения с размытыми краями (в соответствии со схемой 2, см. рис. 2.42) реализован в рамках собственной  $m$ -функции `desmearingf` применительно к варианту 2 и  $m$ -функции `desmearingq` применительно к варианту 4. Об этих функциях уже говорилось выше (на стр. 96).

**Численные иллюстрации.** На рис. 2.43 (см. `text9.m`) и 2.44 (см. `cam8.m`) приведены результаты решения прямой и обратной задач согласно схеме 2 обработки смазанного под углом изображения с использованием размытия краев и двойного поворота изображения. Были использованы следующие операторы:

```
%w - изображение wp на рис. 2.10
%Смазывание изображения:
Delta=40; theta=35;
[g,gir]=smearing(w,Delta,theta,'diffusion');
p=1; %порядок регуляризации
[m,n]=size(w);
w_af=desmearingf(g,Delta,theta,'var2',1e-6,p,'mod',m,n);
[sig_w_af,fshift] = rmsd(w_af,w);
w_aq=desmearingq(g,Delta,theta,'var4',1e-6,'mod',m,n);
[sig_w_aq,qshift] = rmsd(w_aq,w);
%Еще смазывание изображения:
PSF=fspecial('motion',Delta,theta);
c=imfilter(w,PSF,'circular');
Wnrc=deconvwnr(c,PSF,1e-5);
```

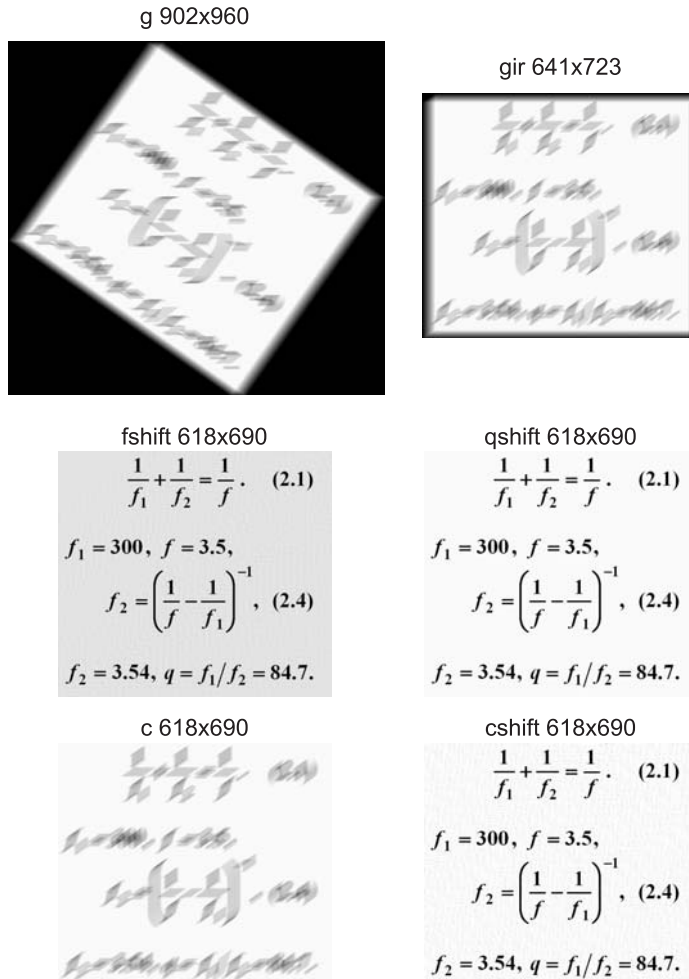


Рис. 2.43. Прямая и обратая задачи обработки смазанного под углом текстового изображения ( $\Delta = 40$  пкс,  $\theta = 35^\circ$ , без шума):

- g — повернутое и смазанное с размытием изображение, схема 2,
- gir — изображение g, повернутое обратно,
- fshift — реконструированное изображение, вариант 2,  $\sigma_{\text{rel}} = 0.1018$ ,
- qshift — реконструированное изображение, вариант 4,  $\sigma_{\text{rel}} = 0.0317$ ,
- c — смазанное изображение, функции `fspecial` и `imfilter`,
- cshift — реконструированное изображение, метод параметрической фильтрации Винера,  $\sigma_{\text{rel}} = 0.0789, 0.0348$  или  $0.0218$





Рис. 2.44. Прямая и обратная задачи обработки смазанного под углом портретного изображения ( $\Delta = 20$  пкс,  $\theta = 35^\circ$ , без шума).

Расшифровка изображений такая же, как на рис. 2.43

```
[sig_Wnrc,cshift] = rmsd(Wnrc,w);
```

Заметим, что, например, изображения `w_af` и `fshift` по внешнему виду практически одинаковы и их отличие состоит в том, что изображение `fshift` немного (не более, чем на 2 пкс) сдвинуто по отношению к изображению `w_af` так, чтобы погрешность `sig_w_af` уменьшилась. Это же относится к изображениям `w_aq` и `qshift`.

Видим, что как и прежде (см. рис. 2.13–2.24) вариант 4 (метод квадратур с размытием краев изображения) дает наиболее точное восстановление изображения. Это подтверждает и рис. 2.44.

**Обработка цветных изображений.** Рассмотрим важный случай, когда изображение является цветным. Этому случаю посвящено много публикаций ([35, 36, 45, 46, 74, 108, 136, 140, 142] и др.).

В пакете IPT системы MatLab цветные изображения представлены в двух видах: в виде RGB (Red, Green, Blue) изображений и в виде индексированных (палитровых, ind) изображений [35, с. 423], [36, с. 206], [45, с. 470], [142, с. 172]. Согласно Международной комиссии по освещению (МКО), цвета красный (R), зеленый (G) и синий (B) полагаются *первичными основными цветами*. *Цветным RGB-изображением* называется массив (3-мерная матрица)  $M \times N \times 3$  из цветных пикселей, причем каждый пиксел содержит три цветовых компонента: красную, зеленую и синюю. Все видимые цвета могут быть в хорошем приближении воспроизведены путем смешения первичных основных цветов в различных пропорциях.

*Индексированные изображения* – это также RGB-изображения, только со специфическим заданием яркостей и цветов (в виде двух 2-мерных матриц – матрицы данных и матрицы цветовой карты map) [36, с. 208], [45, с. 471].

Если в каждом пикселе яркости компонент равны:  $R = G = B$  (но изменяются от пиксела к пикселу), то такое изображение называется *серым* (полутоновым, gray).

Наряду с цветовой системой (цветовым пространством, цветовой моделью) RGB существуют также следующие цветовые системы: CMY – вторичные основные цвета: Cyan (голубой), Magenta (пурпурный) и Yellow (желтый), причем  $C = G + B$ ,  $M = R + B$ ,  $Y = R + G$ ; CMYK – Cyan, Magenta, Yellow и black; HSI – Hue (цветовой тон), Saturation (насыщенность) и Intensity (интенсивность) и др. Однако мы будем рассматривать только систему RGB (и ее частный случай – gray).<sup>7</sup>

Обработка цветных изображений обычно осуществляется следующими двумя способами [35, с. 446], [36, с. 244].

В *первом способе* (способе *покомпонентной обработки*) [109, 140, 168] изображение разбивается на три компонента (цветовые составляющие) R, G и B, решается прямая задача – внесение искажений (смаза и шума) в каждую компоненту, затем решается обратная задача – устранение искажений (реконструкция) каждой компоненты некоторым устойчивым методом, например, методом регуляризации Тихонова или методом параметрической фильтрации Винера и в заключение три компоненты соединяются в единое RGB-изображение, которое можно выполнить с помощью внешней m-функции `cat.m`, имеющейся в пакете IPT системы MatLab.

<sup>7</sup>Тем не менее, при наборе данной книги все цветные изображения (см. цветную вклейку) были переведены из RGB в CMYK.

Проиллюстрируем первый способ на примере обработки цветного RGB-изображения `pears.png` (груши, см. рис. 2.45w).

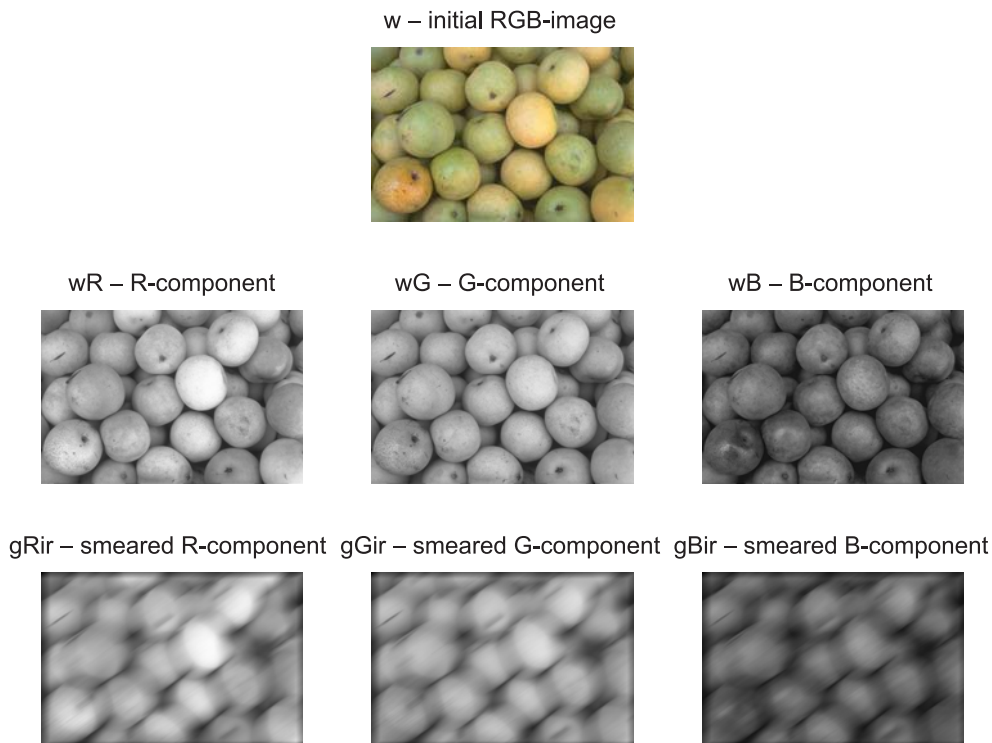


Рис. 2.45. Смазывание цветного изображения первым способом (способом покомпонентной обработки)

Приведем выдержки из головной m-программы `RecSmIm.m`:

```
%RecSmIm.m, головная программа.
%Чтение файла с неискаженным изображением:
w=imread('pears.png'); %486x732x3 uint8
%Вывод на экран (см. рис. 2.45):
figure;
subplot(3,3,2), imshow(w), title('w - initial RGB-image')
%Параметры прямой задачи:
Delta=60; %длина смаза в пкс
theta=41; %угол смаза в град
    %1-й способ
%Выделение трех цветовых составляющих:
wR=w(:,:,1); wG=w(:,:,2); wB=w(:,:,3); %486x732 uint8
[mw,nw]=size(wR); %486,732
%Вывод на экран трех цветовых составляющих:
```

```

subplot(3,3,4), imshow(wR), title('wR - R-component')
subplot(3,3,5), imshow(wG), title('wG - G-component')
subplot(3,3,6), imshow(wB), title('wB - B-component')

%Прямая задача - смазывание с размытием краев:
[gR,gRir]=smearing(double(wR),Delta,theta,'diffusion');
subplot(3,3,7), imshow(uint8(gRir)), title('gRir - smeared R-component')
[gG,gGir]=smearing(double(wG),Delta,theta,'diffusion');
subplot(3,3,8), imshow(uint8(gGir)), title('gGir - smeared G-component')
[gB,gBir]=smearing(double(wB),Delta,theta,'diffusion');
subplot(3,3,9), imshow(uint8(gBir)), title('gBir - smeared B-component')

%Обратная задача - реконструкция составляющих
%методом квадратур и регуляризации Тихонова (вариант 4):
alpha=1e-7;
w_alphaR=desmearingq(gR,Delta,theta,'var4',alpha,'mod',mw,nw);
w_alphaG=desmearingq(gG,Delta,theta,'var4',alpha,'mod',mw,nw);
w_alphaB=desmearingq(gB,Delta,theta,'var4',alpha,'mod',mw,nw);
%СКО и изображения с небольшими сдвигами:
figure; %см. рис. 2.46
[sig_w_aR,wshiftR]=rmsd(w_alphaR,wR); %sig_w_aR=0.0187
subplot(2,3,1), imshow(uint8(wshiftR)), title('wshiftR, \alpha=1e-7')
[sig_w_aG,wshiftG]=rmsd(w_alphaG,wG); %sig_w_aG=0.0199
subplot(2,3,2), imshow(uint8(wshiftG)), title('wshiftG, \alpha=1e-7')
[sig_w_aB,wshiftB]=rmsd(w_alphaB,wB); %sig_w_aB=0.0298
subplot(2,3,3), imshow(uint8(wshiftB)), title('wshiftB, \alpha=1e-7')
%Суммарное СКО (среднее квадратичное) [Бронштейн, Семендяев, с. 139]:
sig_w_a=sqrt((sig_w_aR^2+sig_w_aG^2+sig_w_aB^2)/3); %=0.0233
%Соединение трех составляющих:
w_alphaRGB=cat(3,w_alphaR,w_alphaG,w_alphaB); %485x731x3 double
[sig_w_aRGB,wshiftRGB]=rmsd(w_alphaRGB,w); %sig_w_aRGB=0.0206
subplot(2,3,5), imshow(uint8(wshiftRGB)), title('wshiftRGB', \alpha=1e-7)

```

Во *втором способе* (способе *векторной обработки*) также решаются прямая и обратная задачи, но без разбиения изображения на три компоненты, а именно, изображение обрабатывается как единое целое (цветные пиксели рассматриваются как трехмерные векторы).

Ниже приведены дальнейшие выдержки из головной программы `RecSmIm.m`, посвященные второму способу обработки цветного изображения `pears.png`.

*%2-й способ*

```

%Прямая задача - смазывание с размытием единого RGB-изображения:
[g,gir]=smearing(double(w),Delta,theta,'diffusion');
figure; %см. рис. 2.47

```

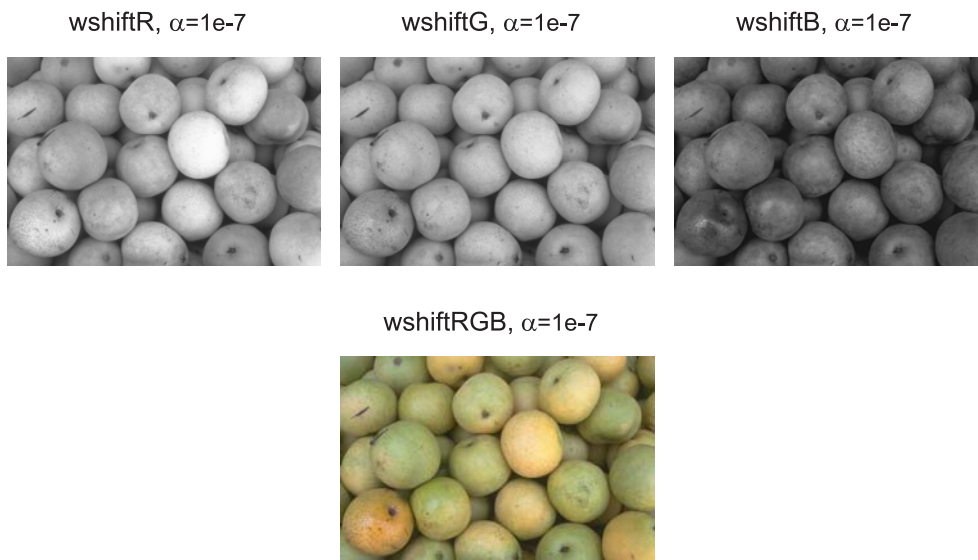


Рис. 2.46. Восстановление (реконструкция) цветного изображения первым способом (способом покомпонентной обработки)

```
subplot(2,2,1), imshow(uint8(gir))
title('gir 525x777x3 - unit smeared RGB-image')
%Обратная задача - реконструкция единого изображения
%методом квадратур и регуляризации Тихонова (вариант 4):
w_alpha=desmearingq(g,Delta,theta,'var4',alpha,'mod',mw,nw);
[sig_w_a,wshift]=rmsd(w_alpha,w); %sig_w_a=0.0206
subplot(2,2,3), imshow(uint8(wshift))
title('wshift 485x731x3, \alpha=1e-7, \sigma_{rel}=0.0206')
%Метод параметрической фильтрации Винера
%Функция рассеяния точки (PSF):
PSF=fspecial('motion',Delta,theta);
%Смазывание изображения:
c=imfilter(w,PSF,'circular'); %(periodic) или
%s=imfilter(w,PSF,'symmetric'); %(reflective)
subplot(2,2,2), imshow(uint8(c))
title('c 486x732x3 - smeared RGB-image')
%Реконструкция изображения:
Wnrc=deconvwnr(c,PSF,1e-3); %K=1e-5 в MatLab7.0.1 и K=1e-3 в MatLab7.10.0
[sig_Wnrc,Wshift]=rmsd(Wnrc,w);
%sig_Wnrc=0.0536 в MatLab7.0.1 и 0.0454 в MatLab7.10.0
subplot(2,2,4), imshow(uint8(Wshift))
title('Wshift 486x732x3, K=1e-3, \sigma_{rel}=0.0454')
```

На рис. 2.45 и 2.46 приведены результаты обработки изображения `pears.png` первым способом, а на рис. 2.47 — результаты обработки вторым способом.

gir 525x777x3 – unit smeared RGB-image    с 486x732x3 – smeared RGB-image



wshift 485x731x3,  $\alpha = 1e-7, \sigma_{rel} = 0.0206$     Wshift 486x732x3,  $K=1e-3, \sigma_{rel} = 0.0454$



Рис. 2.47. Смазывание и восстановление цветного (RGB) изображения вторым способом (способом векторной обработки)

При этом решалась как прямая, так и обратная задача. В прямой задаче выполнялось смазывание с размытием краев цветного изображения под углом по разработанной собственной *m*-функции `smearing.m` с опцией `'diffusion'`. А в обратной задаче выполнялось восстановление (реконструкция) изображения методом квадратур с регуляризацией Тихонова (вариант 4) по разработанной *m*-функции `desmearingq.m`. При этом по разработанной *m*-функции `rmsd.m` рассчитывалось относительное среднеквадратическое отклонение (СКО)  $\sigma_{rel}$  восстановленного изображения от точного.

Следует заметить, что в функции `smearing.m` выполнялся поворот исходного изображения на угол `-theta` (с помощью внешней *m*-функции `imrotate.m`), а затем обратный поворот смазанного изображения на угол `theta`. Кроме того, выполнялось дополнение нулями повернутого на угол `-theta` изображения (с помощью внешней *m*-функции `padarray.m`), а затем – удаление черной рамки вокруг изображения, повернутого обратно на угол `theta`. Все эти операции сопровождалось округлением значений координат точек до ближайших целых значений (с помощью встроенной *m*-функции `round`). Это нередко приводило к сдвигу и смазанного, и восстановленного изображения на величину до 2 пкс как по строкам, так и по столбцам.

В результате, несмотря на малость сдвига, это приводило к завышению (в несколько раз) значения  $\sigma_{\text{rel}}$ . Чтобы устранить эффект сдвига изображения, в `m`-функции `rmsd.m` выполнен подбор небольших сдвигов восстановленного изображения относительно точного до получения минимального значения  $\sigma_{\text{rel}}$ . Эта ситуация напоминает сравнение томограмм, полученных разными типами томографий (РКТ, МРТ, ПЭТ), и это сравнение обычно выполняется за счет введения контрольных точек (точек привязки) [36, с. 204], или характерных (реперных) точек на томограммах [69, с. 108].<sup>8</sup> Однако мы ориентируемся на случай, когда на изображении, вообще говоря, реперных точек нет и поэтому добиваемся наилучшего совмещения восстановленного и точного изображений другим способом, а именно, за счет минимизации  $\sigma_{\text{rel}}$ .

На рис. 2.47 для сравнения приведен также результат смазывания изображения с помощью внешних `m`-функций `fspecial.m` и `imfilter.m`, использующих "граничные условия", и результат восстановления (реконструкции) изображения методом параметрической фильтрации Винера (с помощью внешней `m`-функции `deconvwnr.m`). Видим, что восстановление методом квадратур с регуляризацией Тихонова (вариант 4) лучше, чем методом параметрической фильтрации Винера (значения  $\sigma_{\text{rel}}$  равны соответственно 0.0206 и 0.0536, кроме того, визуально видно, что на изображении `Wshift` присутствуют волны Гиббса, а на `wshift` их практически нет). Это еще раз говорит о пользе введения размытия краев у изображения (ср. `gir` и `s` на рис. 2.47).

Заметим при этом, что исходное изображение `pears.png` искажалось только смазыванием, но на него не накладывался шум. Это было сделано специально, чтобы четче выявить роль смазывания под углом, размытия краев и трехцветности.

Работа первого и второго способов обработки цветных изображений, а также работа собственных `m`-функций `smearing.m`, `desmearingq.m` и `rmsd.m` (а также `desmearingf.m` и `normnoise.m`) была проверена, помимо `pears.png`, на других цветных изображениях (`football.jpg`, `onion.png`, `peppers.png`, `saturn.png`, `flower.bmp` и др.).

На титульной обложке книги приведено смазанное RGB-изображение мяча (файл `football.jpg`), выполненное с помощью `m`-функции `smearing.m`, и результат его реконструкции методом квадратур и регуляризации Тихонова (вариант 4) по `m`-функции `desmearingq.m` (2-й способ).

**Обработка реальных смазанных изображений.** Практика показывает, что существует заметное различие между решением модельных примеров (обработка фантомов) и обработкой реальных (натурных) искаженных изображений. Обработка реальных изображений заметно сложнее обработки модельных изображений и это можно объяснить следующими причинами.

При обработке реальных изображений возникают значительные трудности с определением функции рассеяния точки (ФРТ) [26, 27, 82]. При этом следует различать два случая: 1-й случай, когда мы располагаем лишь одним искаженным изображением и не можем провести дополнительный эксперимент с целью определения ФРТ, и 2-й случай, когда можно провести специальный эксперимент для определения ФРТ или рассчитать ФРТ. 2-й случай имел место, например, в ситуации с космическим

<sup>8</sup>О реперных (опорных, сопряженных) точках (в задаче мониторинга местности) см. также в [40, с. 108].

телескопом "Хаббл" (КТХ, HST) (см. п. 2.2) [135, 173]. В этой ситуации, во-первых, имелась возможность определить экспериментально ФРТ (путем наведения КТХ на одиночную звезду) и, во-вторых, можно было рассчитать ФРТ (поскольку причина искажения изображений космических объектов была известна – остаточная сферическая абберрация зеркала телескопа).

Если же имеет место 1-й случай, то параметры ФРТ, например, величину и направление смаза  $\Delta$  можно оценить по штрихам на изображении (см. `gir` и `s` на рис. 2.47) или путем подбора, но это обычно выполняется с погрешностью.

Кроме того, на изображение, помимо смаза, накладывается шум, характеристики которого (МО, СКО, закон распределения и т.д.) также трудно оценить. В этом отношении ситуация с КТХ была благоприятной, поскольку за пределами атмосферы, в вакууме внешний шум практически отсутствует. Поэтому неудивительно, что множество искаженных изображений, полученных КТХ, удалось реконструировать с высокой точностью. В 1-м же случае мы имеем довольно мало результатов хорошей реконструкции реальных смазанных изображений: [4, с. 165], [92, 120, 140] и др.

Была выполнена обработка *реального смазанного цветного изображения* портретного типа. На рис. 2.48(1) (см. `real_pair.m`) представлен исходный фотоснимок (файл `pair.jpg`) — смазанное RGB-изображение двух персон (пары).

Из анализа изображения `rgb` и путем решения обратной задачи при различных значениях величины смаза  $\Delta$  и угла смаза  $\theta$  были выбраны следующие их наиболее вероятные значения:  $\Delta = 32$  пкс,  $\theta = -8^\circ$ . На рис. 2.48(2) приведено изображение `r` — результат поворота изображения `rgb` на угол  $-\theta$ . Далее изображение `r` реконструировалось методом конечных сумм (квадратур) с регуляризацией Тихонова по `m`-файлу `desmearingq.m` и для сравнения методом параметрической фильтрации Винера по `m`-функции `deconvwnr.m`. При этом значения параметра регуляризации  $\alpha$  и константы  $K$  были подобраны, исходя из визуальной оценки получаемых изображений `w_alpha` и `Wnr`, равными  $\alpha = K = 1.4 \cdot 10^{-1}$ . При таких значениях  $\alpha$  и  $K$  были получены реконструированные изображения `w_alpha` и `Wnr` (см. рис. 2.48(3) и (4)).

Можно отметить следующее. На смазанном изображении `rgb` (а также `r`) сильно искажены лица, а на реконструированных изображениях `w_alpha` и `Wnr` лица просматриваются довольно хорошо, но значителен ореол вокруг каждой персоны. Поэтому была выполнена дополнительная обработка изображений `w_alpha` и `Wnr` с помощью графических редакторов PaintBrush и PhotoShop, а именно, устранен (посредством "кисти") ореол, а также выполнена равномерная заливка фона. При этом сами изображения (лица, головы, туловища) этой обработке не подвергались. На рис. 2.48(5) и (6) — окончательный результат обработки. Видим, что полученные изображения `w_alpha_cor` и `Wnr_cor` являются заметно более четкими (хотя и далеко неидеальными), чем исходное искаженное изображение `rgb`.

Рассмотрим реконструкцию еще одного реального смазанного под углом цветного изображения — фотографии здания ратуши в центре Стокгольма, Швеция.<sup>9</sup> Эта фотография была получена в 1991 г. на аналоговом пленочном фотоаппарате "Зенит" и хранилась в виде диапозитива. Фотография была переведена в электронный `jpg`-файл

<sup>9</sup>Обработка данного изображения была выполнена аспирантом Р.А. Экземпляровым.



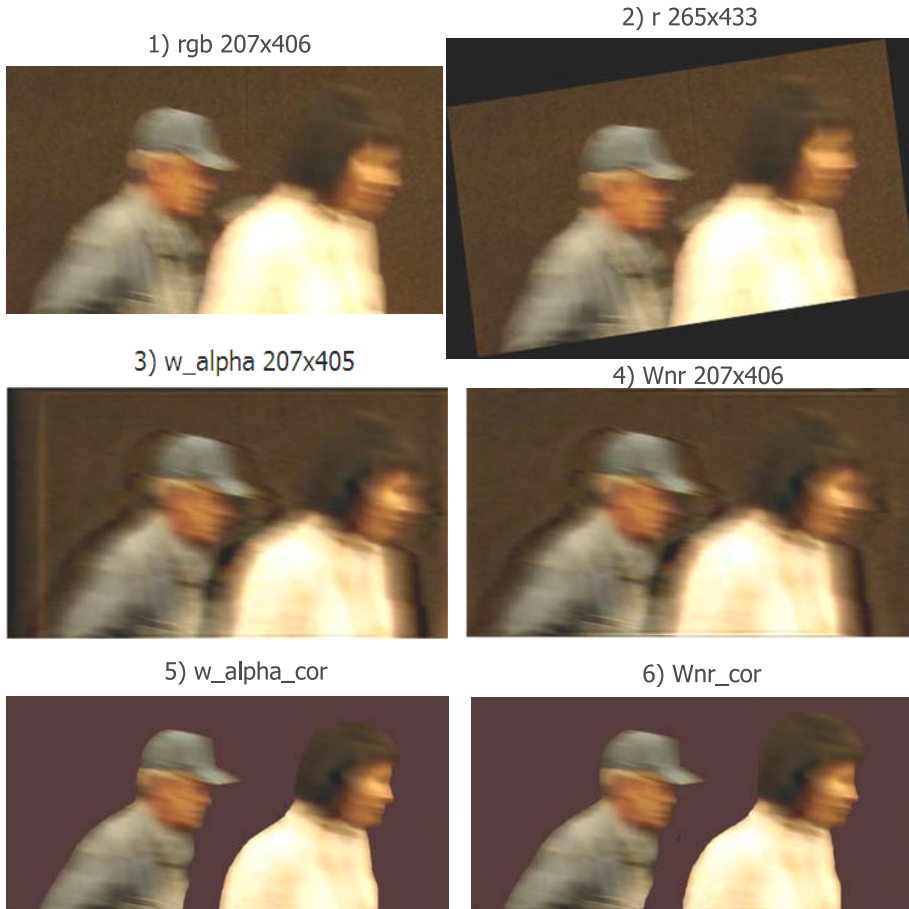


Рис. 2.48. Обработка реального портретного изображения

размера  $2081 \times 3159 \times 3$ , а затем этот файл был переведен (с помощью PhotoShop'a) в файл `rehoitus1.jpg` меньшего размера  $659 \times 1000 \times 3$  (см. рис. 2.49).

Из рис. 2.49 видно, что изображение покрыто штрихами, царапинами, точками, пылью и т.п. Эти артефакты были устранены, но не математическим путем, а с помощью редактора PhotoShop (см. `rehoitus2.jpg`  $638 \times 1000 \times 3$  на рис. 2.50а).

После устранения артефактов видно (из рис. 2.50а), что на снимке присутствует смаз. Для устранения смазывания была использована параметрическая фильтрация Винера через посредство  $m$ -функций `fspecial.m` и `deconvwnr.m` с подобранными параметрами  $\Delta = 15$ ,  $\theta = 88$ ,  $K = \text{NSR} = 0.05$ .

На рис. 2.50б (см. `Rehoitus.m`) — результат реконструкции изображения, а на рис. 2.50в и 2.50г — увеличенные варианты реконструированного изображения. Из рис. 2.50г видно, что стала читаемой надпись на здании: "REHOITUS" (первые две буквы закрыты флагом на мачте), что означает в переводе со шведского "Финансы". На исходной фотографии (рис. 2.49) надпись прочитать невозможно. Отметим важный



Рис. 2.49. Реальное смазанное цветное изображение  
(файл rehoitus1.jpg 659 × 1000 × 3)

факт, что в настоящее время данной надписи на здании нет, т.е. нам удалось восстановить историческую информацию путем математико-компьютерной обработки снимка.

Отметим также, что изображение реконструировалось недостаточно четко и это можно объяснить наличием эффекта Гиббса и тем, что функция рассеяния точки (ФРТ) не является пространственно-инвариантной (разностной): при внимательном рассмотрении здания на рис. 2.49 можно увидеть, что его левая часть смазана чуть сильнее, чем правая, т.е. при фотографировании фотоаппарат не только сдвинулся прямолинейно (почти вертикально), но и испытал некоторое вращение.

Сказанное можно отнести и к изображению на рис. 2.48: персоны двигались с разными скоростями, в результате чего ФРТ получилась неразностной, т.е. величина смаза  $\Delta \neq \text{const}$ . Эти тонкие эффекты при реконструкции нами не были учтены и это повысило погрешность реконструкции.

### Контрольные задания и вопросы

1. Что такое встроенные, внешние и собственные m-функции (или m-файлы) системы MatLab?

2. Объясните записи заголовков функций:

```
function [g,gir] = smearing(w,Delta,theta,type)
```

и

```
function w_alpha = desmearingf(g,Delta,theta,variant,alpha,occasion,m,n)
```

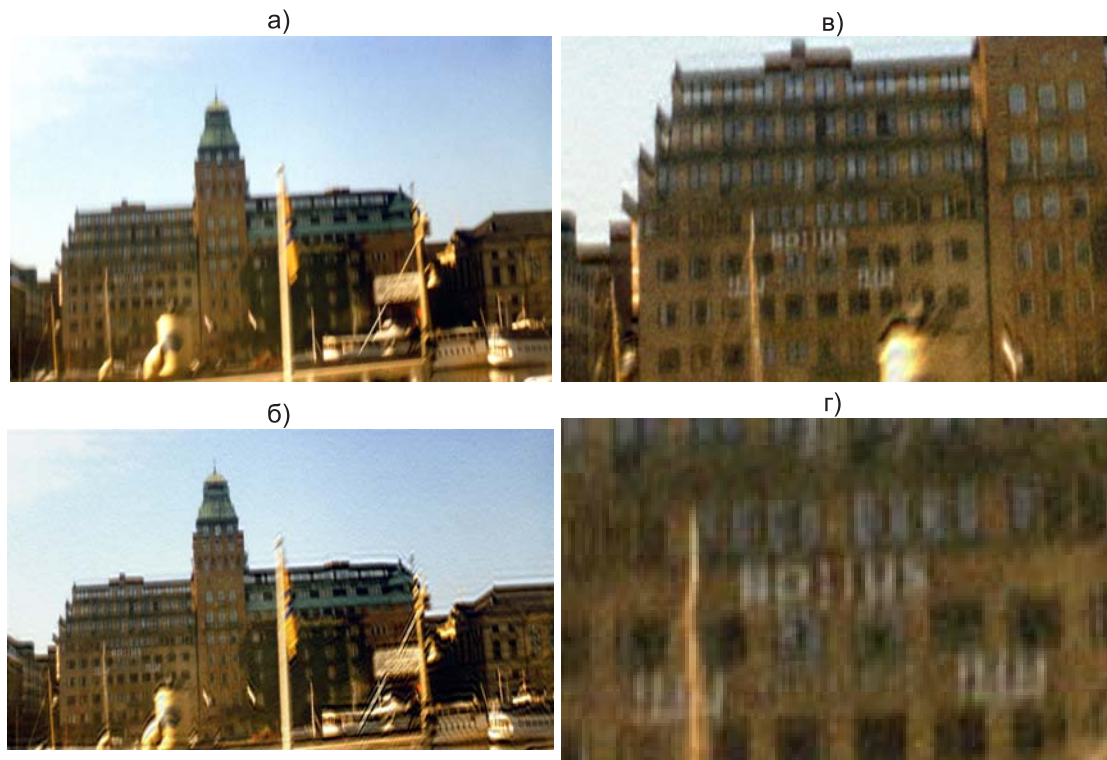


Рис. 2.50. Процесс реконструкции изображения rehoitus1.jpg:  
 а — изображение после устранения артефактов,  
 б — изображение после реконструкции с помощью  
 функций `fspecial.m` и `deconvwnr.m`,  
 в и г — увеличенные б-изображения

Запишите возможные варианты обращений к этим функциям.

3. Укажите специфику обработки текстового, космического и портретного изображений. В чем состоят особенности обработки модельного и реального изображений?

4. Изложите два способа выбора параметра регуляризации  $\alpha$  и параметра (константы)  $K$  в методах регуляризации Тихонова и параметрической фильтрации Винера, которые используются в данной книге.

5. В чем состоит преимущество обработки негатива по сравнению с обработкой позитива при использовании приема усечения изображения? Почему метод параметрической фильтрации Винера дает ложные волны (эффект Гиббса) на восстановленном изображении (рис. 2.30 и 2.33(6s))?

6. Почему погрешность  $\sigma_{rel}(\alpha)$  (рис. 2.18) имеет повышенные значения при малых и больших значениях  $\alpha$  и имеет пониженные значения при "умеренных"  $\alpha$ ? Нужно дать объяснение, обращаясь к формулам (2.28), (2.29), (2.38) метода регуляризации Тихонова.

7. Объясните, почему на рис. 2.21(3) изображение `w_a4pi` имеет наибольшую по-

грешность реконструкции  $\sigma_{\text{rel}}$ , хотя это негатив.

8. Почему функция `edgetaper` практически не устраняет искажения типа "звоны" (эффект Гиббса)?

9. Дайте определения гауссова и импульсного шумов. В чем состоит идея соединения фильтрации (шума) и регуляризации? Какая  $m$ -функция фильтрует шум адаптивным фильтром Винера и какая — медианным фильтром? Изложите кратко суть этих фильтров.

10. Почему даже небольшая погрешность в величине  $\Delta$  при решении обратной задачи методом регуляризации Тихонова может привести к значительной погрешности реконструкции изображения (ср. изображения на рис. 2.21(1) и 2.27(1))?

11. На основе анализа таблицы сделайте заключения о зависимости погрешности реконструкции  $\sigma_{\text{rel}}$  от уровня шума и его типа для различных методов реконструкции и о том, помогает ли дополнительная фильтрация адаптивным фильтром Винера и медианным фильтром понизить погрешность  $\sigma_{\text{rel}}$ .

12. Изложите идею быстрого алгоритма реконструкции смазанных изображений (см. (2.47)–(2.49)).

13. В чем состоят особенности алгоритма смазывания изображения под произвольным углом  $\theta$ , реализованного во внешних  $m$ -функциях `fspecial` и `imfilter`? В чем заключается идея "приема поворота" (см. рис. 2.41–2.44)?

14. Охарактеризуйте основные виды цветных изображений. Что такое серое изображение? Изложите два способа обработки цветных изображений — способы покомпонентной и векторной обработки.

15. В чем состоят различия в обработке модельных и реальных смазанных изображений?

## 2.5. Реконструкция дефокусированных изображений

Рассмотрим еще одну важную задачу обработки изображений — задачу реконструкции (восстановления, реставрации) дефокусированных изображений (фотоснимков человека, текста, природы, телескопических снимков космических объектов — планет, туманностей, галактик, снимков наземных объектов со спутников, изображений в микроскоп биологических объектов — вирусов, ДНК и т.д.) [2, 4, 6, 17, 18, 26–28, 35, 36, 39, 40, 46, 60, 73, 81, 82, 91, 92, 102, 105, 106, 108, 109, 120, 131, 133–137, 140, 142, 143, 148, 150, 166–168, 173–175].

Дефокусирование изображения может возникнуть вследствие неправильной установки фокуса фотоаппарата (будет иметь место расфокусировка) или в результате прохождения изображения через атмосферу с турбулентностью [35, с. 384] (возникнет нечеткое изображение) или из-за наличия сферической аберрации у зеркала телескопа, как это имело место у космического телескопа "Хаббл" [135, 173] (в этом случае будет иметь место эффект, аналогичный расфокусировке).

Рассмотрим прямую задачу (моделирование дефокусирования) и обратную задачу (устранение дефокусирования математическим путем) на примере фотоаппарата.

**Постановка задачи.** Задача дефокусирования имеет много общего с предыдущей задачей (смазывание изображений), но имеет и существенные отличия.

Пусть снимаемый объект (полагаемый плоским из-за его обычной удаленности) и фото пленка расположены параллельно тонкой линзе по разные стороны от нее. Обозначим через  $f_1$  расстояние от линзы до объекта, а через  $f_2$  то расстояние от линзы до фото пленки (идеальной фото пленки, установленной в "фокусе"), которое удовлетворяет соотношению (2.1) и при котором на фото пленке будет четкое изображение объекта. Однако мы полагаем, что расстояние от линзы до фото пленки равно не  $f_2$ , а  $f_2 + \delta$ , где  $\delta$  — погрешность фокусировки изображения, т.е. реальная фото пленка установлена "не в фокусе" (см. рис. 2.51).

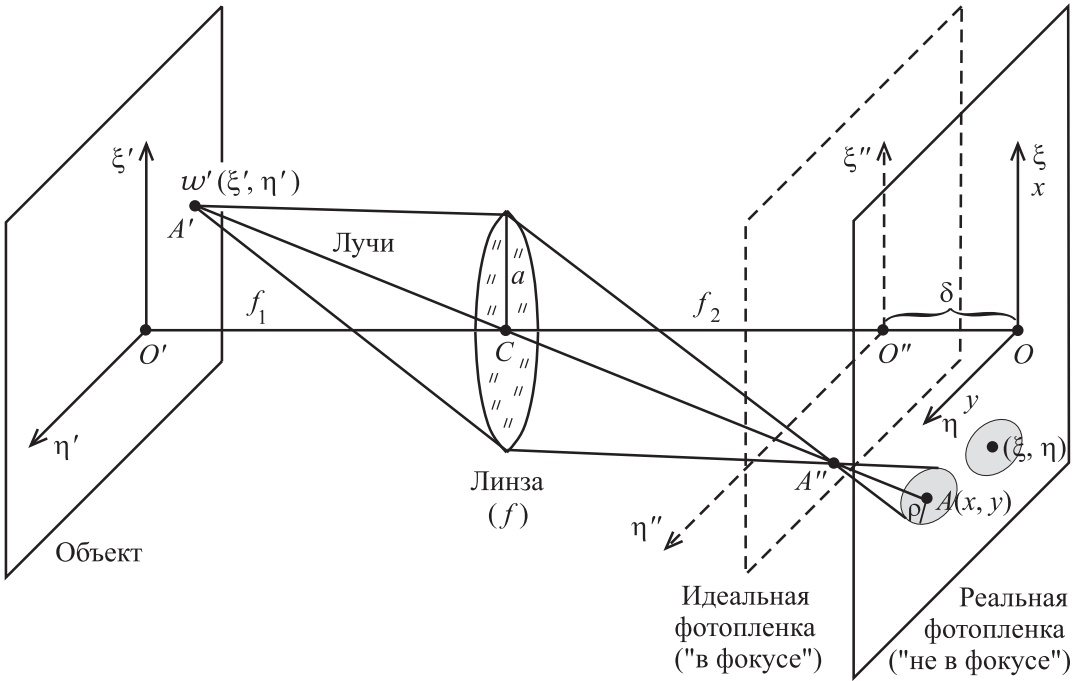


Рис. 2.51. Схема получения дефокусированного изображения

Введем в плоскости объекта прямоугольную систему координат  $\xi'O'\eta'$ , на "идеальной" фото пленке — систему координат  $\xi''O''\eta''$ , а на реальной фото пленке — систему координат  $\xi O\eta$ , а также совпадающую с ней  $xOy$ . Обозначим через  $w'(\xi', \eta')$  интенсивность, исходящую из некоторой точки  $A'(\xi', \eta')$  объекта. Лучи из точки  $A'$  после их прохождения через линзу пересекутся на "идеальной" фото пленке также в точке  $A''$  с координатами  $\xi'' = -\xi'/q, \eta'' = -\eta'/q$ , где  $q = f_1/f_2$  (ср. (2.2)), причем  $f_2$  определяется согласно (2.3). При этом интенсивность в точке  $A''$  будет равна  $w''(\xi'', \eta'') = w'(\xi', \eta')$ .

На реальной же фото пленке точка  $A'$  отобразится не в точку, а в некоторое размытое пятно, являющееся ничем иным как *функцией рассеяния точки* — ФРТ (point spread function — PSF). Сначала рассмотрим простейший случай, когда каждое такое пятно (т.е. ФРТ) является однородным кругом радиуса  $\rho$  [172, р. 385], что соответствует пространственно-инвариантной ФРТ (на рис. 2.51 отобразен именно этот

простейший случай). Заметим, что во внешней  $m$ -функции `fspecial.m` пакета IPT системы MatLab 7 этому случаю соответствует вариант: `h=fspecial('disk',radius)`.

На рис. 2.52 приведен пример дефокусированного изображения, полученного из исходного изображения `textp.bmp` (см. рис. 2.10wp) следующим образом:

```
RGBp=imread('textp.bmp'); b=uint8(5); wp=double(rgb2gray(RGBp)-b);
PSF=fspecial('disk',10); defoc=imfilter(wp,PSF,'symmetric');
figure, imshow(uint8(defoc)), title('defoc 618x690,\rho=10')
```

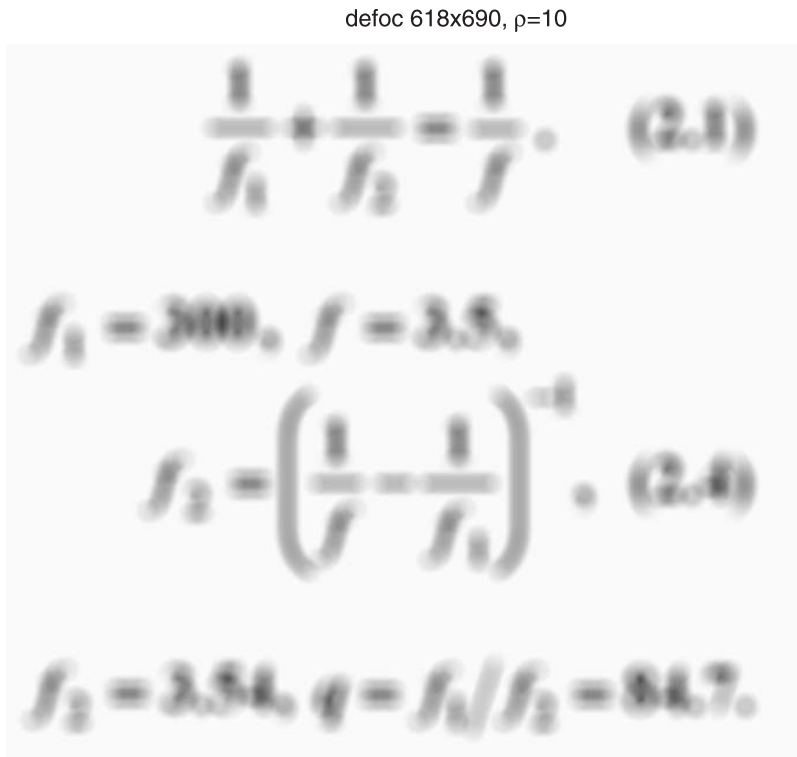


Рис. 2.52. Пример дефокусированного изображения (простейший случай, ФРТ — однородный круг)

Видно, что точки на изображении (всего пять точек) действительно превратились в однородные круги одинакового радиуса.

**Вывод основного соотношения в простейшем случае.** Опишем математически задачу дефокусирования в простейшем случае. В этом случае точка  $A'$  отобразится на реальной фотоплёнке в однородный круг радиуса

$$\rho = \frac{a \delta}{f_2} \quad (2.51)$$

с центром в точке  $A(x, y)$ , где  $a$  — радиус апертуры линзы<sup>10</sup>, причем

$$x = -\frac{f_2 + \delta}{f_1} \xi', \quad y = -\frac{f_2 + \delta}{f_1} \eta'. \quad (2.52)$$

Рассмотрим, помимо круга с центром в точке  $A(x, y)$ , также некоторый другой круг с центром в точке  $(\xi, \eta)$  (см. рис. 2.51). Радиусы этих (а также других) кругов одинаковы (так как рассматривается случай пространственно-инвариантной ФРТ) и равны  $\rho$  (см. (2.51)), а площади кругов равны  $S = \pi\rho^2$ . В результате интенсивность  $w(\xi, \eta)$ , соответствующая точке  $(\xi, \eta)$ , будет "размазана" по кругу радиуса  $\rho$  и площади  $S = \pi\rho^2$  с постоянной плотностью интенсивности  $w(\xi, \eta)/\pi\rho^2$ .

Интенсивность в точке  $A(x, y)$  будет результатом суммирования (интегрирования) по всем тем кругам, которые накрывают точку  $A(x, y)$ . Условие накрытия точки  $A(x, y)$  кругом с центром в точке  $(\xi, \eta)$  и радиусом  $\rho$  есть

$$\sqrt{(x - \xi)^2 + (y - \eta)^2} \leq \rho. \quad (2.53)$$

В результате интенсивность в точке  $A(x, y)$  будет равна

$$g(x, y) = \iint_{\sqrt{(x-\xi)^2+(y-\eta)^2} \leq \rho} \frac{w(\xi, \eta)}{\pi\rho^2} d\xi d\eta. \quad (2.54)$$

Соотношение (2.54) является основным в *прямой задаче* обработки дефокусированного изображения (задаче моделирования дефокусирования) в простейшем случае (когда ФРТ — однородный круг постоянного радиуса).

Если соотношение (2.54) видоизменить, лишь поменяв местами левую и правую части, то получим принципиально иное соотношение, а именно, *интегральное уравнение*

$$\iint_{\sqrt{(x-\xi)^2+(y-\eta)^2} \leq \rho} \frac{w(\xi, \eta)}{\pi\rho^2} d\xi d\eta = g(x, y). \quad (2.55)$$

Интегральное уравнение (2.55) является основным в *обратной задаче* обработки дефокусированного изображения (задаче реконструкции) в простейшем случае.

Несмотря на схожесть записей соотношений (2.54) и (2.55), они имеют принципиальное различие. Если в соотношении (2.54) задается  $w$  — неискаженное изображение и рассчитывается (моделируется)  $g$  — дефокусированное изображение путем вычисления интеграла (прямая задача), то в уравнении (2.55) задается дефокусированное изображение  $g$  (которое может быть смоделировано, а может быть получено и в реальном эксперименте), а определяется искомое изображение  $w$  путем решения интегрального уравнения (обратная задача). При этом обратная задача является значительно более сложной, чем прямая задача — в этом мы убедились при рассмотрении задачи о смазанных изображениях (п. 2.3 и 2.4) и убедимся далее.

<sup>10</sup> *Апертура* — действующее отверстие оптической системы [128, с. 32].

**Приведение основного уравнения к стандартной форме.** Уравнение (2.55) есть двухмерное интегральное уравнение I рода относительно  $w(\xi, \eta)$ . Однако оно имеет *нестандартную форму*: вместо пределов интегрирования стоит неравенство (2.53), к тому же в явном виде нет ядра (точнее, оно равно константе  $1/\pi\rho^2$ ). В этом отношении уравнение (2.55) напоминает уравнение Радона (1.6).

Нестандартность уравнения (2.55) создает сложности с его решением. Преобразуем его к стандартной форме. Запишем (2.55) в виде [2], [4, с. 165], [40, с. 82], [102, с. 75], [106, 108, 111, 120], [168, р. 200]:

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x - \xi, y - \eta) w(\xi, \eta) d\xi d\eta = g(x, y), \quad -\infty < x, y < \infty, \quad (2.56)$$

где

$$h(x - \xi, y - \eta) = \begin{cases} \frac{1}{\pi\rho^2}, & \sqrt{(x - \xi)^2 + (y - \eta)^2} \leq \rho, \\ 0, & \text{иначе,} \end{cases} \quad (2.57)$$

или

$$h(x, y) = \begin{cases} \frac{1}{\pi\rho^2}, & \sqrt{x^2 + y^2} \leq \rho, \\ 0, & \text{иначе.} \end{cases} \quad (2.58)$$

Соотношение (2.56) есть двухмерное интегральное уравнение Фредгольма I рода типа свертки. В нем  $g(x, y)$  — интенсивность на дефокусированном фотоснимке,  $w(\xi, \eta)$  — искомая интенсивность на неискаженном снимке (на снимке, который был бы получен при  $\delta = 0$ ),  $h(x, y)$  есть ядро интегрального уравнения, являющееся функцией рассеяния точки (ФРТ) в простейшем случае, когда ФРТ — однородный круг постоянного радиуса  $\rho$ .

При этом значение  $\rho$  можно оценить согласно (2.51), где  $a$  и  $f_2$  известны, а  $\delta$  можно подобрать (подобрать такое значение  $\delta$ , при котором реконструкция изображения будет выполнена наиболее успешно). Однако целесообразнее подбирать непосредственно  $\rho$  без использования формулы (2.51). Например, по изображению, приведенному на рис. 2.52, значение  $\rho$  можно оценить как радиус наименьших деталей (в данном случае, это радиус кругов, в которые превратились точки).

Отметим, что в работах [4, с. 166], [120] рассмотрена также задача реконструкции дефокусированного изображения в случае наклонного расположения объекта (когда плоскости объекта и фотоплёнки непараллельны).

Наряду с простейшим случаем, когда ФРТ имеет вид (2.58), рассмотрим более реальный случай, когда ФРТ имеет вид *гауссианы* [2], [40, с. 82], [108]:

$$h(x, y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}. \quad (2.59)$$



В этом случае каждая точка на объекте превращается в пятно с интенсивностью, убывающей от центра к краям. Дефокусирование изображения с ФРТ в виде гауссианы можно смоделировать с помощью внешних m-функций `fspecial` и `imfilter`. На рис. 2.53 приведен пример дефокусированного изображения, полученного с помощью этих функций:

```
RGBp=imread('textp.bmp'); %618x690x3 uint8
b=uint8(5); %фон
wp=double(rgb2gray(RGBp)-b); %618x690 double
PSF=fspecial('gaussian',[26 26],5); %26x26 double
defoc=imfilter(wp,PSF,'symmetric'); %618x690 double
figure, imshow(uint8(defoc)), title('defoc 618x690, \sigma=5')
```

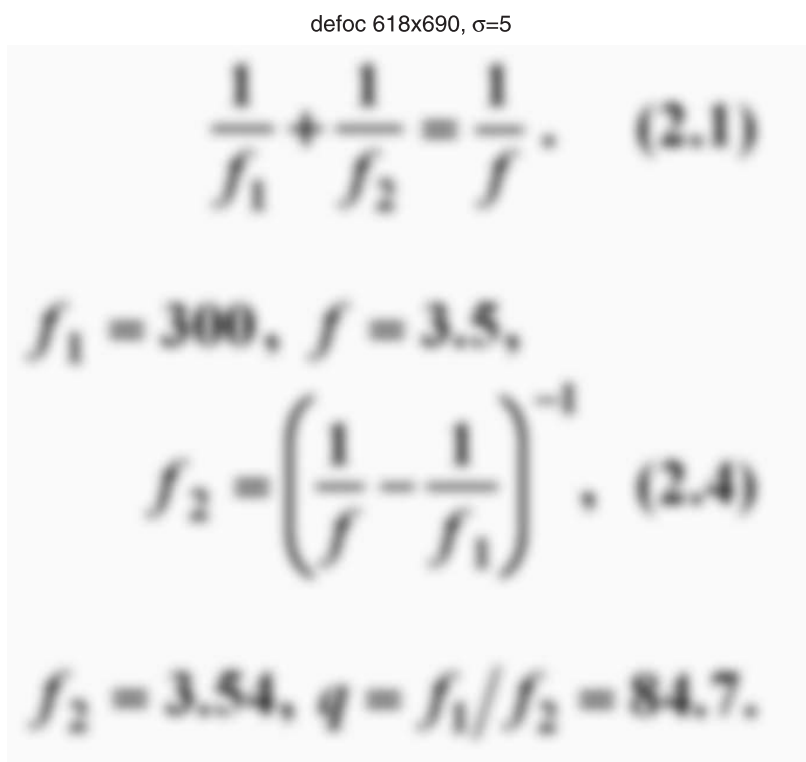


Рис. 2.53. Пример дефокусированного изображения (ФРТ — гауссиана)

Можно несколько упростить задачу, если воспользоваться известным правилом "трех сигм":

$$h(x, y) = \begin{cases} \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}, & \sqrt{x^2 + y^2} \leq 3\sigma, \\ 0, & \text{иначе.} \end{cases} \quad (2.60)$$

Это приведет к тому, что в  $m$ -функции `fspecial('gaussian', [H H], sigma)` можно полагать:  $H = 6\sigma$ , и это приведет к сокращению вычислительных операций и к ускорению обработки. Здесь  $H$  — ширина квадратного окна с гауссианой внутри него. К этому вопросу мы еще вернемся при изложении численных алгоритмов.

Однако ФРТ может иметь более общий вид, чем (2.58) или (2.59) ([39, 82, 150, 167, 179] и др.). В этом случае, если ФРТ является разностной, или пространственно-инвариантной:  $h(x - \xi, y - \eta)$ , т.е. она одинакова во всех точках изображения, то задача дефокусирования по-прежнему описывается уравнением (2.56), но  $h(x, y)$  будет иметь вид, вообще говоря, отличный от (2.58) или (2.59). Если же ФРТ является пространственно-зависимой [91, с. 102]:  $h(x, \xi, y, \eta)$ , то задача дефокусирования будет описываться более общим уравнением [39], [66, с. 18]:

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x, \xi, y, \eta) w(\xi, \eta) d\xi d\eta = g(x, y). \quad (2.61)$$

Соотношение (2.61) есть двухмерное интегральное уравнение Фредгольма I рода с общим ядром.

Наглядные примеры устройств наблюдения, у которых ФРТ (или аппаратная функция — АФ, или характеристика направленности — ХН) является пространственно-инвариантной, или разностной, — это фотоаппарат, глаз, ухо и т.д., а устройства, у которых ФРТ (или АФ или ХН) зависит от  $(x, y)$  — это, например, антенны подводных лодок, у которых поворот ХН осуществляется электрически (за счет внесения задержек между преобразователями) [102, с. 117].

**Численные алгоритмы решения прямой задачи.** Под *прямой задачей* будем подразумевать моделирование дефокусирования изображения. Прямая задача сводится к вычислению  $g(x, y)$  — распределению интенсивности по дефокусированному изображению при заданном  $w(\xi, \eta)$  — распределении интенсивности по неискаженному изображению и заданной функции рассеяния точки (ФРТ)  $h(x, y)$  или  $h(x, \xi, y, \eta)$  согласно

$$g(x, y) = \iint_D h(x - \xi, y - \eta) w(\xi, \eta) d\xi d\eta, \quad x, y \in F, \quad (2.62)$$

или

$$g(x, y) = \iint_D h(x, \xi, y, \eta) w(\xi, \eta) d\xi d\eta, \quad x, y \in F. \quad (2.63)$$

Здесь интегрирование ведется по  $D$  — области исходного изображения  $w$  (FOV — field of view) (ср. п. 2.3). Через  $F$  обозначена область дефокусированного изображения  $g$ .

Для выполнения численных расчетов необходимо алгоритм в непрерывной (интегральной) форме (2.62) или (2.63) записать в дискретной форме (см. дальше). При этом, как было уже отмечено в п. 2.3, в работах ([35, 36, 144, 148–150, 167, 178] и др.) часто используются так называемые "*граничные условия*" (boundary conditions, BCs), а также в работе [143, с. 103–107] *способы доопределения* для определения неизвестных (недостающих) значений  $w$  вне заданной ограниченной области  $D$  (или FOV).

"Граничные условия" являются по-существу способами экстраполяции значений  $w$  за границы  $B$  области  $D$ . Поэтому их логичнее называть *внеграничными условиями*.

"Граничные условия" бывают следующие: zero Dirichlet, symmetric (reflective, Neumann), replicate, circular (periodic) и anti-reflective. Например, anti-reflective (наиболее эффективное ВС) заключается в том, что значения  $w$  вне границы (края)  $B$  вычисляются путем зеркального отражения относительно края и последующего зеркального отражения относительно нормали к краю. Первое условие (zero) целесообразно применять в случае, когда функцию  $w$  можно считать финитной, т.е.  $w = 0$  вне  $D$ , например, в случае космического объекта [167, Figure 4], а остальные условия — когда  $w$  не является финитной, т.е. носитель изображения в действительности шире заданной области  $D$ . "Граничные условия" нашли отражение во внешней  $m$ -функции `imfilter.m` пакета ИРТ системы MatLab 7 через посредство опций '`symmetric`', '`replicate`', '`circular`' и др. Способы доопределения (дополнение нулями, периодическое и четное продолжение) [143, с. 105] весьма напоминают "граничные условия".

Непрерывным соотношениям (2.62) и (2.63) можно поставить в соответствие дискретные соотношения вида:

$$g = Aw, \quad (2.64)$$

где  $A$  — матрица, связанная с ФРТ и с "граничными условиями". Как уже отмечалось в п. 2.3, например, в случае ВС reflective матрица  $A$  есть сумма блочно-теплицевой матрицы с теплицевыми блоками (ВТТВ) и блочно-ганкелевой матрицы с ганкелевыми блоками (ВННВ) [167]. Как и в п. 2.3, можно сделать заключение, что введение ВСs является довольно искусственным приемом и, кроме того, их использование порождает сложный математический аппарат.

В данном пункте мы изложим более простое решение этой проблемы, которое, как и в п. 2.3, будем называть *приемом (способом) усечения изображения*.

**Прием усечения дефокусированного изображения.** Во внешней функции `fspecial.m` для случая дефокусирования реализовано три варианта моделирования ФРТ типа '`average`', '`disk`' и '`gaussian`' и во всех вариантах используется окно (квадратная матрица), внутрь которого помещается однородный круг (в варианте '`disk`') или гауссиана (в варианте '`gaussian`'). При изложении нижеследующего приема усечения мы поступим аналогичным образом, а именно, будем полагать, что функция рассеяния точки  $h$  может быть в принципе любого вида ((2.58), (2.59), (2.60) и т.д.), но в пределах квадратного окна размером  $2r \times 2r$  или матрицы  $(2r+1) \times (2r+1)$  пкс (см. рис. 2.54).

Рассмотрим прямую задачу моделирования дефокусирования изображения в дискретном виде [108]. Пусть дана матрица исходного неискаженного изображения  $w_{m \times n}$  и нужно смоделировать матрицу дефокусированного изображения  $g$ . Полагаем при этом, что функция рассеяния точки  $h$  — любого вида, но пространственно-инвариантная (разностная) и заключенная в пределах квадратного окна  $2r \times 2r$  (рис. 2.54), а вне окна равная нулю, т.е.  $h$  описывается матрицей  $(2r+1) \times (2r+1)$ . Если формировать дефокусированное изображение  $g$  такого же размера, как и исходное изображение  $w$ , т.е. формировать  $g$  размером  $m \times n$ , то соотношение (2.64) более подробно запишется

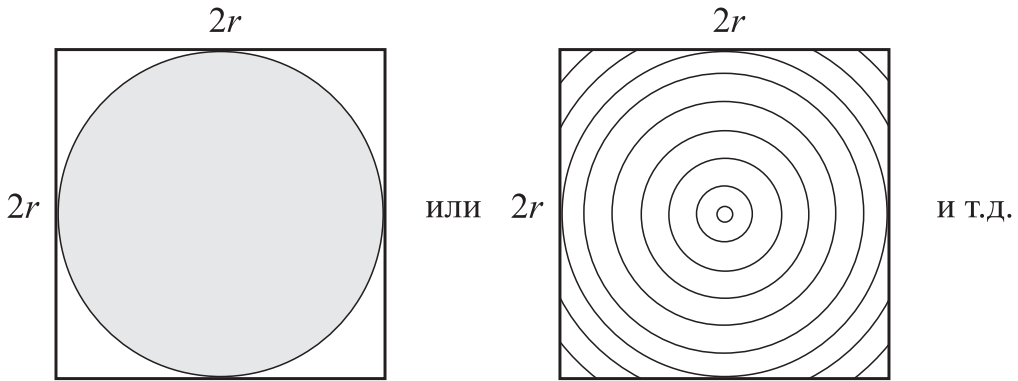


Рис. 2.54. Различные виды ФРТ в пределах квадратного окна

в следующем виде (если заменить интеграл в (2.62) конечной суммой по пикселям):<sup>11</sup>

$$g(j, i) = \frac{1}{Q} \sum_{j'=j-r}^{j+r} \sum_{i'=i-r}^{i+r} h(i-i', j-j') w(j', i'), \quad j = 1, \dots, m, \quad i = 1, \dots, n, \quad (2.65)$$

где  $j$  (и  $j'$ ) — номер строки,  $i$  (и  $i'$ ) — номер столбца. В соотношении (2.65) выражение для ФРТ  $h(i, j)$  нужно записывать несколько иначе, чем (2.57)–(2.60), а именно,<sup>12</sup>

$$h(i, j) = \begin{cases} 1, & \sqrt{i^2 + j^2} \leq r, \\ 0, & \text{иначе,} \end{cases} \quad (2.58')$$

$$h(i, j) = e^{-(i^2+j^2)/2\sigma^2}. \quad (2.59')$$

Это связано с тем, что в (2.65) введен нормирующий делитель  $Q$ , равный

$$Q = \sum_{i=-r}^r \sum_{j=-r}^r h(i, j). \quad (2.66)$$

В результате, во-первых, запись (2.59') и запись  $h(i, j) = (1/2\pi\sigma^2)e^{-(i^2+j^2)/2\sigma^2}$  дадут одинаковый численный результат для  $g(j, i)$  (2.65), а во-вторых, например, при  $r = 0$  (т.е. в отсутствие дефокусирования) получим:  $g(j, i) = w(j, i)$ .

На рис. 2.55 приведена схема расчета  $g(j, i)$  согласно (2.65). Квадратами  $2r \times 2r$  отмечены различные возможные положения квадратного окна, внутри которого заключена ФРТ  $h$ .

<sup>11</sup>Напоминаем, что согласно Соглашению о координатах (см. стр. 70) в непрерывном случае мы пишем:  $g(x, y)$ , а в дискретном:  $g(j, i)$ , где  $x$  и  $i$  (а также  $y$  и  $j$ ) имеют одинаковое направление.

<sup>12</sup>При написании программы в системе MatLab необходимо  $g$  и  $w$  оформлять как матрицы, а  $h$  — как функцию.

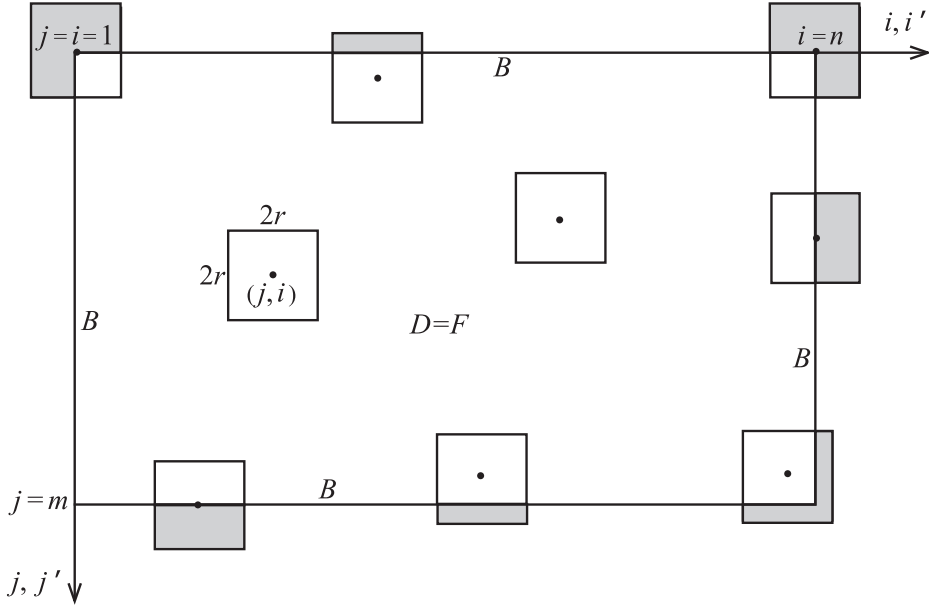


Рис. 2.55. Схема расчета дефокусированного изображения  $g$  по формуле (2.65).  $B$  — граница  $D$  (области  $w$ ) и  $F$  (области  $g$ )

Проанализируем выражение (2.65). Видим, что при  $i = n - r + 1, \dots, n$  (вблизи правой границы  $B$  на рис. 2.55) требуется знание значений  $w(j, n - 2r + 1), \dots, w(j, n + r)$ , однако значения  $w(j, n + 1), \dots, w(j, n + r)$  (за правой границей  $B$ ) неизвестны. Аналогично требуются, но неизвестны значения  $w$  ниже, левее и выше границы  $B$  на величину  $r$  пикселей. На рис. 2.55 затемнены те части кадров, в которых неизвестны значения  $w$ . Чтобы преодолеть этот недостаток информации, в цикле работ ([5, 6, 45, 59, 60, 62, 92, 102, 103, 113] и др.), как мы уже отмечали, предложены и используются так называемые "граничные условия": от простейшего нуля до anti-reflective, в которых делаются различные предположения относительно недостающих значений  $w$ . Заметим, что такой эффект "недостачи" имеет место только при обработке модельных изображений в прямой задаче. Что же касается реальных дефокусированных изображений, то при их получении сама природа предоставляет необходимые значения  $w$  в дефокусированное изображение  $g$ .

Мы предлагаем иное решение вопроса, связанного с недостатком значений  $w$ , а именно, вместо соотношения (2.65) использовать соотношение (см. рис. 2.56) [108]:

$$\hat{g}(j, i) = \frac{1}{Q} \sum_{j'=j-r}^{j+r} \sum_{i'=i-r}^{i+r} h(i-i', j-j') w(j', i'),$$

$$j = 1 + r, \dots, m - r, \quad i = 1 + r, \dots, n - r. \quad (2.67)$$

Отличие выражения (2.67) от (2.65) состоит в том, что согласно (2.67) моделируется изображение  $\hat{g}$  меньшего размера, чем  $w$ . При этом все необходимые значения

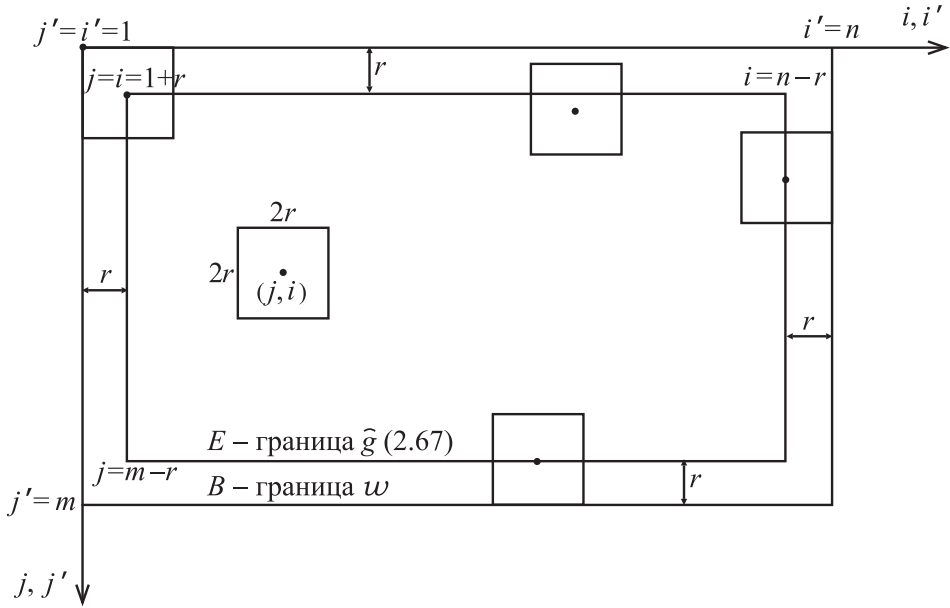


Рис. 2.56. Схема расчета дефокусированного изображения с усечением согласно (2.67)

$w(j', i')$ ,  $j' = 1, 2, \dots, m$ ,  $i' = 1, 2, \dots, n$ , известны и не нужно прибегать к такому приему, как "граничные условия". Правда, изображение  $\hat{g}$  получается несколько меньшего размера, чем  $w$  (на  $r$  пикселей сверху, снизу, слева и справа). Но, во-первых, практически эффект уменьшения невелик (обычно  $m \approx n \approx 300 \div 1000$ , а  $r \sim 10$ ), а, во-вторых, при решении обратной задачи (задачи реконструкции), например, методом регуляризации Тихонова или методом итеративной регуляризации Фридмана и т.д. в качестве искомого решения  $w_\alpha$  используется нормальное псевдорешение [22, с. 236], [117, с. 112], которое существует и является единственным, несмотря на то, что задача отыскания  $w$  по  $\hat{g}$  является недоопределенной (область  $F$  изображения  $\hat{g}$  уже области  $D$  изображения  $w$ ).

Заметим, что выражение (2.67) не вполне удобно для программирования в системе MatLab, поскольку в системе MatLab начальные индексы у массивов должны равняться единице. Более удобным является выражение (см. рис. 2.57) [108]:

$$\hat{g}(j, i) = \frac{1}{Q} \sum_{j'=j}^{j+2r} \sum_{i'=i}^{i+2r} h(i - i' + r, j - j' + r) w(j', i'),$$

$$j = 1, \dots, m - 2r, \quad i = 1, \dots, n - 2r, \quad (2.68)$$

которое получено заменой в правой части (2.67)  $j$  на  $j + r$  и  $i$  на  $i + r$ . При этом еще раз отметим, что при программировании в системе MatLab следует полагать  $g$  и  $w$  двумерными массивами (матрицами), а  $h$  —  $m$ -функцией.

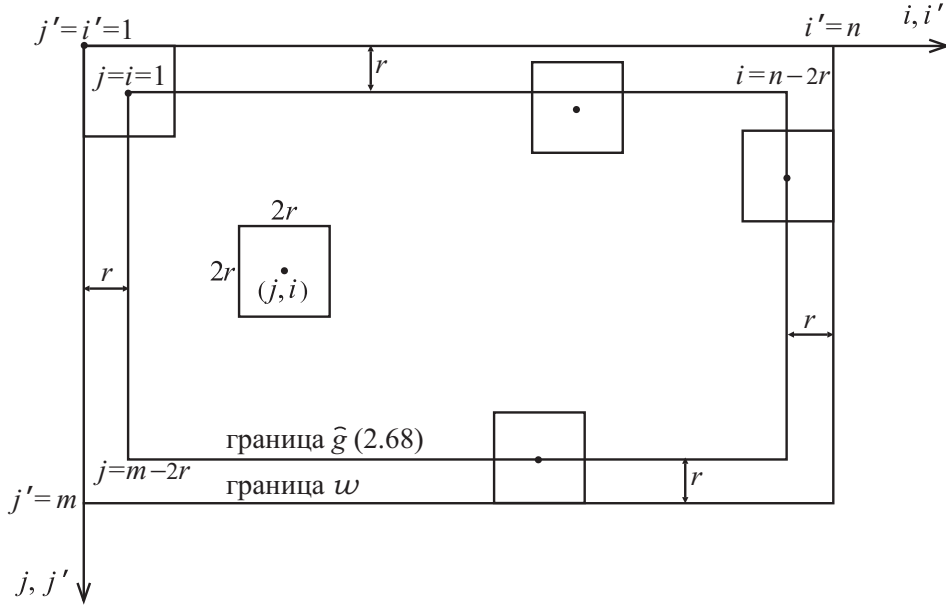


Рис. 2.57. Схема расчета дефокусированного изображения с усечением согласно (2.68)

Однако и выражение (2.68) является недостаточно удобным для программирования в системе MatLab, поскольку в нем фигурирует два массива (матрицы)  $\hat{g}$  и  $w$  и функция  $h$ , а MatLab демонстрирует свою максимальную эффективность, когда он оперирует исключительно с матрицами. Поэтому введем  $M$  — матрицу на основе функции  $h$  следующим образом [108]:

$$M(j, i) = h(i - r - 1, j - r - 1), \quad j = 1, \dots, 2r + 1, \quad i = 1, \dots, 2r + 1. \quad (2.69)$$

Тогда (2.68) запишется в виде [106, 108]:

$$\hat{g}(j, i) = \frac{1}{Q} \sum_{j'=j}^{j+2r} \sum_{i'=i}^{i+2r} M(j - j' + 2r + 1, i - i' + 2r + 1) w(j', i'),$$

$$j = 1, \dots, m - 2r, \quad i = 1, \dots, n - 2r. \quad (2.70)$$

Заметим, что в выражении (2.70)  $\min(j - j' + 2r + 1) = \min(i - i' + 2r + 1) = 1$ , а  $\max(j - j' + 2r + 1) = \max(i - i' + 2r + 1) = 2r + 1$ .

Назовем моделирование дефокусированного изображения  $\hat{g}$  на основе (2.67)–(2.70) *приемом усечения дефокусированного изображения* (по аналогии с приемом усечения смазанного изображения, изложенным в п. 2.3).

**Прием размытия краев дефокусированного изображения.** В п. 2.3 и 2.4 мы уже убедились, что при решении обратной задачи (задачи реконструкции) на восстанавливаемом изображении (в основном, на его краях) нередко возникают искажения





$$j = 1, \dots, m + 2r, \quad i = 1, \dots, n + 2r, \quad (2.71)$$

где  $u$  — матрица  $(m+4r) \times (n+4r)$ , введенная для удобства расчетов путем добавления к матрице  $w$  по  $2r$  нулей слева, справа, сверху и снизу (это можно сделать с помощью функции `padarray.m`).

**Собственные m-функции для моделирования дефокусирования изображений.** Разработано несколько m-функций в системе MatLab для моделирования дефокусирования изображений. Основная функция — это `defocusing.m`, заголовок и инструкция к которой приведены ниже. Заметим, что если изображение является цветным (RGB), то оно обрабатывается способом векторной обработки (см. стр. 148).

```
function g = defocusing(w,r,type,h) %собственная m-функция
%Моделирование дефокусированного gray- или RGB-изображения
%с его усечением согласно (2.69), (2.70) (type='truncation')
%или с размытием его краев согласно (2.71) (type='diffusion').
%Это - прямая задача.
%Входные параметры (аргументы):
%w - матрица изображения типа gray или RGB размера mхn класса double,
%r - полуширина квадратного окна в пкс (double),
%   в которое должна быть вписана ФПТ (PSF) h,
%type - если type='truncation', то моделируется дефокусированное
%       изображение с усечением, а если type='diffusion',
%       то с размытием краев,
%h - функция рассеяния точки (ФПТ, PSF), которая
%    должна быть оформлена в виде m-функции h(i,j,r).
%Выходной параметр (аргумент):
%g - дефокусированное gray- или RGB-изображение класса double
%    размера (m-2r)х(n-2r), если type='truncation', или
%    размера (m+2r)х(n+2r), если type='diffusion',
%    g должно быть такого же типа (gray или RGB), как и w.
%Пример обращения к defocusing:
%g = defocusing(w,5,'diffusion',@h_disk);
```

С помощью функции `defocusing.m` было выполнено дефокусирование ряда изображений-фантомов (прямая задача).

Сначала было обработано *текстовое изображение*, дефокусированное, но без шума (см. первую часть головной программы `RecDefImTt.m`):

```
%RecDefImTt.m (первая часть)
RGBp = imread('textp.bmp'); %618x690x3 uint8 (позитив)
w = double(rgb2gray(RGBp)-uint8(5)); %618x690 double (фон=5)
figure; %Fig. 2.59:
subplot(2,2,1), imshow(uint8(w)), title('w 618x690')
```

```

%Прямая задача - дефокусирование изображения:
r = 10; %полуширина квадратного окна (см. рис. 2.54)
g1 = defocusing(w,r,'truncation',@h_disk); %598x670 double
subplot(2,2,2), imshow(uint8(g1)), title('g1 598x670')
g2 = defocusing(w,r,'diffusion',@h_disk); %638x710 double
subplot(2,2,3), imshow(uint8(g2)), title('g2 638x710')
%Для сравнения:
PSF = fspecial('disk',r); %ФРТ 21x21 double
%defoc = imfilter(w,PSF,'symmetric'); %618x690 double
%или
defoc = imfilter(w,PSF,'full'); %638x710 double
subplot(2,2,4), imshow(uint8(defoc)), title('defoc 638x710')

```

В приведенной программе `RecDefImTt.m` дефокусирование с помощью собственной *m*-функции `defocusing` выполнено с усечением согласно (2.69), (2.70) (получено изображение `g1`) и с размытием краев изображения согласно (2.71) (получено изображение `g2`). Кроме того, для сравнения выполнено дефокусирование с помощью внешних *m*-функций `fspecial` и `imfilter` с использованием "граничного условия" `'symmetric'` или опции `'full'` (получено изображение `defoc`). При этом использована ФРТ в виде однородного диска радиуса  $r$  (см. рис. 2.54, формулу (2.58') и *m*-функцию `h_disk.m`):

```

function PSF = h_disk(i,j,r) %m-функция
if (sqrt(i^2+j^2) <= r)
    PSF = 1; %точка (i,j) - внутри круга радиуса r
else
    PSF = 0; %точка (i,j) - вне круга
end

```

На рис. 2.59 представлены результаты дефокусирования серого текстового изображения (прямая задача). Результаты реконструкции дефокусированного изображения (обратная задача) будут представлены дальше.

Было обработано также *цветное* (RGB) *изображение* – см. первую часть головной программы `RecDefImTm1.m`), где рассматривается обработка дефокусированного, зашумленного импульсным шумом изображения.<sup>13</sup>

```

%RecDefImTm1.m (первая часть)
RGB = imread('flower.bmp'); %218x248x3 uint8
w = double(RGB); %218x248x3 double
%Прямая задача - дефокусирование изображения:
r = 5; %полуширина квадратного окна (см. рис. 2.54)
%Дефокусирование с усечением:
g1 = defocusing(w,r,'truncation',@h_disk); %208x238x3 double

```

<sup>13</sup>На CD см. также программу `RecDefImTm1.m`, где шум полагается гауссовым.

<p>w 618x690</p> $\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$ <p><math>f_1 = 300, f = 3.5,</math></p> $f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$ <p><math>f_2 = 3.54, q = f_1/f_2 = 84.7.</math></p>	<p>g1 598x670</p> $\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$ <p><math>f_1 = 300, f = 3.5,</math></p> $f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$ <p><math>f_2 = 3.54, q = f_1/f_2 = 84.7.</math></p>
<p>g2 638x710</p> $\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$ <p><math>f_1 = 300, f = 3.5,</math></p> $f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$ <p><math>f_2 = 3.54, q = f_1/f_2 = 84.7.</math></p>	<p>defoc 638x710</p> $\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$ <p><math>f_1 = 300, f = 3.5,</math></p> $f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$ <p><math>f_2 = 3.54, q = f_1/f_2 = 84.7.</math></p>

Рис. 2.59. Дефокусирование серого изображения:

w — исходное изображение,

g1 — дефокусированное изображение с усечением согласно (2.70),

g2 — дефокусированное изображение с размытием краев согласно (2.71),

defoc — дефокусированное изображение с использованием функций `fspecial` и `imfilter` (опция 'full')

```
%Добавление импульсного шума к g1:
rand('state',0); %установка генераторов случайных
randn('state',0); %чисел в начальное состояние
d = 0.02; %доля зашумленных пикселей
g1n = double(imnoise(uint8(g1),'salt & pepper',d)); %208x238x3
%Дефокусирование с размытием краев:
g2 = defocusing(w,r,'diffusion',@h_disk); %228x258x3 double
%Добавление импульсного шума к g2:
g2n = double(imnoise(uint8(g2),'salt & pepper',d)); %228x258x3 double
%Для сравнения
%дефокусирование изображения с использованием
%"граничных условий" symmetric и circular:
PSF = fspecial('disk',r); %ФПТ 21x21 double
defocs = imfilter(w,PSF,'symmetric'); %(reflective) 218x248x3
```

```

defoccc = imfilter(w,PSF,'circular'); %(periodic) 218x248x3
%Dобавление импульсного шума к defoccs и defoccc:
defoccsn = double(imnoise(uint8(defoccs),'salt & pepper',d)); %218x248x3
defocccn = double(imnoise(uint8(defoccc),'salt & pepper',d)); %218x248x3
figure; %Fig. 2.60:
subplot(2,2,1), imshow(uint8(g1)), title('g1 208x238x3')
subplot(2,2,2), imshow(uint8(g2)), title('g2 228x258x3')
subplot(2,2,3), imshow(uint8(defoccs)), title('defoccs 218x248x3')
subplot(2,2,4), imshow(uint8(defoccc)), title('defoccc 218x248x3')
figure; %Fig. 2.61:
subplot(2,2,1), imshow(uint8(g1n)), title('g1n 208x238x3')
subplot(2,2,2), imshow(uint8(g2n)), title('g2n 228x258x3')
subplot(2,2,3), imshow(uint8(defoccsn)), title('defoccsn 218x248x3')
subplot(2,2,4), imshow(uint8(defocccn)), title('defocccn 218x248x3')

```

На рис. 2.60 представлены результаты дефокусирования цветного изображения (прямая задача), а на рис. 2.61 – результат добавления цветного импульсного шума с помощью  $m$ -функции `imnoise`.

**Решение обратной задачи реконструкции дефокусированных изображений.** Под *обратной задачей* подразумевается задача реконструкции (восстановления, реставрации) изображения  $w$  по измеренному дефокусированному изображению  $g$  и известной (некоторым образом) функции рассеяния точки (ФРТ, PSF)  $h$ .

Если ФРТ пространственно-инвариантная (разностная):  $h = h(x - \xi, y - \eta)$ , то задача реконструкции сводится к решению двумерного ИУ Фредгольма I рода типа свертки (2.56). Наиболее распространенные методы решения уравнения (2.56) — это методы инверсной фильтрации, псевдоинверсной фильтрации, оптимальной и параметрической фильтрации Винера, регуляризации Тихонова, алгоритмы Люси-Ричардсона, "слепой" деконволюции и т.д. [2, 4, 18, 27, 35, 36, 39, 40, 91, 102, 108, 109, 112, 120, 137, 150, 167, 168, 172]. Эти методы используют в той или иной мере двумерное преобразование Фурье.

Существуют еще алгебраические, или матричные методы и итерационные методы, использующие квадратуры (точнее, кубатуры), например, итеративный метод минимизации сглаживающего функционала Тихонова (использующий метод сопряженных градиентов) [167], метод итеративной регуляризации Фридмана [22, с. 272] и др.

Мы кратко изложим методы инверсной и псевдоинверсной фильтрации, оптимальной и параметрической фильтрации Винера, регуляризации Тихонова и итеративной регуляризации Фридмана (см. также [102, гл. 8]).

В *методе инверсной фильтрации* [35, с. 388], [36, с. 183], или *двухмерного преобразования Фурье* [102, с. 76] решение уравнения (2.56) записывается в виде двумерного обратного преобразования Фурье (ОПФ) (ср. (2.21)–(2.24)):

$$w(\xi, \eta) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} W(\omega_1, \omega_2) e^{-i(\omega_1 \xi + \omega_2 \eta)} d\omega_1 d\omega_2, \quad (2.72)$$



Рис. 2.60. Дефокусирование цветного (RGB) изображения (без шума):  
 g1 — дефокусирование с усечением согласно (2.70) и `defocusing`,  
 g2 — дефокусирование с размытием краев согласно (2.71) и `defocusing`,  
 defocs и defocc — дефокусирование с использованием функций  
`fspecial` и `imfilter` (опции `'symmetric'` и `'circular'`)

где ПФ (спектр) решения

$$W(\omega_1, \omega_2) = \frac{G(\omega_1, \omega_2)}{H(\omega_1, \omega_2)}, \quad (2.73)$$

причем

$$G(\omega_1, \omega_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y) e^{i(\omega_1 x + \omega_2 y)} dx dy \quad (2.74)$$

— ПФ (спектр) дефокусированного изображения  $g(x, y)$ ,

$$H(\omega_1, \omega_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x, y) e^{i(\omega_1 x + \omega_2 y)} dx dy \quad (2.75)$$

— ПФ (спектр, оптическая передаточная функция — ОПФ, Optical Transfer Function — OTF) функции рассеяния точки. С учетом шума в  $g(x, y)$  выражение (2.73) следует

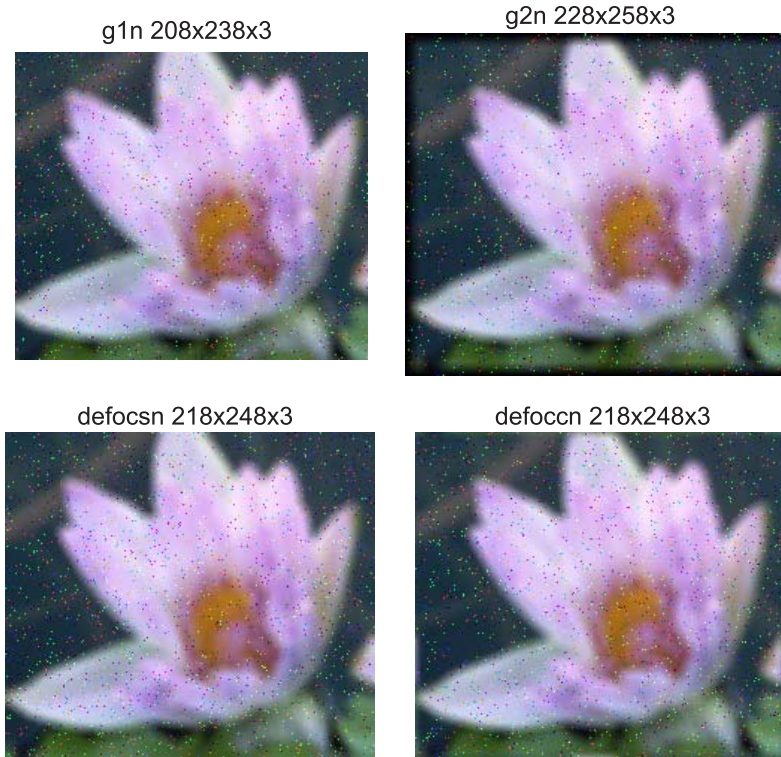


Рис. 2.61. Добавление импульсного цветного шума с помощью  $m$ -функции `imnoise` к дефокусированным цветным изображениям `g1`, `g2`, `defocsn` и `defoccn` (см. рис. 2.60)

записать в виде:

$$\widetilde{W}(\omega_1, \omega_2) = \frac{G(\omega_1, \omega_2) + N(\omega_1, \omega_2)}{H(\omega_1, \omega_2)}, \quad (2.76)$$

где  $N(\omega_1, \omega_2)$  — спектр шума  $n(x, y)$ .

Функция рассеяния точки  $h(x, y)$  нередко задается в виде аналитической формулы типа (2.58)–(2.60), поэтому в принципе ОПФ  $H(\omega_1, \omega_2)$  может быть найдена аналитически согласно (2.75). А  $G(\omega_1, \omega_2)$  должно быть найдено численно с помощью функции `fft2.m` (двухмерное БПФ) с шумом. При этом метод инверсной фильтрации оказывается неустойчивым. Во-первых, функция  $H(\omega_1, \omega_2)$  может принимать нулевые или близкие к нулевым значения при некоторых  $\omega_1, \omega_2$ , вследствие чего даже в отсутствие шума функция  $W(\omega_1, \omega_2)$  может иметь нежелательные "выбросы". Во-вторых, при наличии шума  $H(\omega_1, \omega_2) \rightarrow 0$  при  $\omega_1, \omega_2 \rightarrow \infty$ , а  $N(\omega_1, \omega_2) \rightarrow \text{const}$  при  $\omega_1, \omega_2 \rightarrow \infty$ , где `const` есть "белый шум". Это аналогично тому, что имеет место в методе инверсной фильтрации решения одномерного уравнения (2.13).

Отмеченная неустойчивость метода инверсной фильтрации обусловлена тем, что задача решения уравнения (2.56) является *некорректной*.

Различные модификации метода инверсной фильтрации называются *псевдоинверсной фильтрацией* [36, с. 183]. Например, учитывается, что на практике  $\omega_{1\max}$  и  $\omega_{2\max}$  — конечные величины (частоты Найквиста), поэтому в (2.72) пределы интегрирования будут конечными и это в некоторой степени понизит неустойчивость решения (ср. с фильтром Рамачандрана–Лакшминараяна, см. (1.16); см. (2.26) — решение в пространстве  $S_{\pi/h}$  [22, с. 259]; см. также [35, с. 389] — использование низкочастотного фильтра Баттерворта и т.д.). Однако псевдоинверсная фильтрация недостаточно эффективна, так как довольно грубо отсекает высокие частоты Фурье, которые, с одной стороны, наиболее чувствительны к шумам и поэтому их необходимо подавлять, но, с другой стороны, они нужны для повышения разрешения. Гораздо более аккуратно подавляют высокие частоты методы фильтрации Винера и регуляризации Тихонова.

В *методе оптимальной фильтрации Винера* [17, с. 105–113], [22, с. 315–316], [35, с. 390–392], [36, с. 183–186], [102, с. 210–211], [117, с. 201], [172, с. 382–383] решение уравнения (2.56) ищется исходя из условия минимума величины  $E[w_R(\xi, \eta) - w(\xi, \eta)]^2$ , где  $w_R(\xi, \eta)$  — решение уравнения (2.56) (реконструированное изображение),  $w(\xi, \eta)$  — точное решение (неискаженное изображение), а  $E$  — оператор математического ожидания по ансамблю реализаций. В результате

$$w_R(\xi, \eta) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} W_R(\omega_1, \omega_2) e^{-i(\omega_1 \xi + \omega_2 \eta)} d\omega_1 d\omega_2, \quad (2.77)$$

где

$$W_R(\omega_1, \omega_2) = \frac{H^*(\omega_1, \omega_2) G(\omega_1, \omega_2)}{|H(\omega_1, \omega_2)|^2 + R_n(\omega_1, \omega_2)/R_w(\omega_1, \omega_2)}, \quad (2.78)$$

$R_n(\omega_1, \omega_2)$  и  $R_w(\omega_1, \omega_2)$  — спектральная плотность мощности шума и неискаженного изображения соответственно или, в случае одной реализации шума,  $R_n(\omega_1, \omega_2) = |N(\omega_1, \omega_2)|^2$  — энергетический спектр шума и  $R_w(\omega_1, \omega_2) = |W(\omega_1, \omega_2)|^2$  — энергетический спектр неискаженного изображения.

При этом среднеквадратическое отклонение решения  $w_R$  от точного  $w$  равно [22, с. 316], [102, с. 211]

$$E[w_R(\xi, \eta) - w(\xi, \eta)]^2 = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{R_n(\omega_1, \omega_2)}{|H(\omega_1, \omega_2)|^2 + R_n(\omega_1, \omega_2)/R_w(\omega_1, \omega_2)} d\omega_1 d\omega_2. \quad (2.79)$$

Формула (2.77) дает соотношение для реконструированного дефокусированного изображения методом фильтрации Винера, а формула (2.79) — значение квадрата погрешности реконструкции, являющейся минимально возможной — поэтому фильтр называется *оптимальным*.

Однако метод оптимальной фильтрации Винера трудно (или вообще невозможно) реализовать на практике. Это связано с тем, что спектр шума  $R_n(\omega_1, \omega_2)$  с некоторым приближением можно определить, а спектр неискаженного изображения  $R_w(\omega_1, \omega_2)$  в большинстве случаев неизвестен [35, с. 392].

В этом случае часто отношение  $R_n(\omega_1, \omega_2)/R_w(\omega_1, \omega_2)$  заменяют на некоторую константу (параметр)  $K$ , в результате чего (2.77), (2.78) можно записать в виде [35, с. 392], [36, с. 184], [108], [172, р. 382–383]:

$$W_K(\omega_1, \omega_2) = \frac{H^*(\omega_1, \omega_2) G(\omega_1, \omega_2)}{|H(\omega_1, \omega_2)|^2 + K}, \quad (2.80)$$

$$w_K(\xi, \eta) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} W_K(\omega_1, \omega_2) e^{-i(\omega_1\xi + \omega_2\eta)} d\omega_1 d\omega_2 \quad (2.81)$$

и такой вариант метода называется *методом параметрической фильтрации Винера* [36, с. 184]. Винеровская фильтрация реализована в системе MatLab 7 в виде внешней m-функции `deconvwnr.m`.

Если обрабатывается модельное изображение, когда  $w(\xi, \eta)$  известно и в принципе можно использовать (2.77), (2.78), но, тем не менее, используются (2.80), (2.81), то параметр  $K$  можно определить по минимуму относительной погрешности реконструкции  $\sigma_{\text{rel}}(K)$  — см. (2.44). Если же обрабатывается реальное дефокусированное изображение, то параметр  $K$  наиболее удобно определить визуальным способом, выводя на экран (дисплей, монитор) реконструированные изображения  $w_K(\xi, \eta)$  при различных значениях  $K$  и делая выбор  $K$  на основе визуального восприятия изображения.

В *методе регуляризации Тихонова* (в варианте, использующем двухмерное ПФ) [2], [4, с. 165], [22, с. 268–269], [26, 27], [35, с. 397], [37, с. 146–147], [38, с. 37–38], [102, с. 202–204], [108, 120], [121, с. 43–44], [140], [168, р. 201] решение уравнения (2.56) получается из условия минимума сглаживающего функционала

$$\|Aw - g\|^2 + \alpha \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} M(\omega_1, \omega_2) |W(\omega_1, \omega_2)|^2 d\omega_1 d\omega_2 = \min_w, \quad (2.82)$$

где  $A$  — интегральный оператор левой части уравнения (2.56), а  $M(\omega_1, \omega_2)$  — регуляризатор, например,

$$M(\omega_1, \omega_2) = (\omega_1^2 + \omega_2^2)^p, \quad (2.83)$$

$p \geq 0$  — порядок регуляризации. В результате (ср. (2.28), (2.29))

$$w_\alpha(\xi, \eta) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} W_\alpha(\omega_1, \omega_2) e^{-i(\omega_1\xi + \omega_2\eta)} d\omega_1 d\omega_2, \quad (2.84)$$

где

$$W_\alpha(\omega_1, \omega_2) = \frac{H^*(\omega_1, \omega_2) G(\omega_1, \omega_2)}{|H(\omega_1, \omega_2)|^2 + \alpha(\omega_1^2 + \omega_2^2)^p}. \quad (2.85)$$

Дадим краткое сравнение методов инверсной фильтрации, параметрической фильтрации Винера и регуляризации Тихонова. Если в (2.80) параметр  $K$  равен нулю, то метод параметрической фильтрации Винера переходит в метод инверсной фильтрации — неустойчивый метод. Параметр  $K$  при удачном (ненулевом) выборе создает и



устойчивость реконструкции изображения  $w$ , и приемлемую точность реконструкции. Если в (2.85) параметр  $p$  равен нулю, то метод регуляризации Тихонова переходит в метод параметрической фильтрации Винера. Однако при  $p > 0$  (обычно  $p \in [0.5, 2]$ ) метод регуляризации Тихонова становится более эффективным, так как слагаемое  $\alpha(\omega_1^2 + \omega_2^2)^p$  умеренно подавляет высокие Фурье-частоты  $\omega_1$  и  $\omega_2$ , в которых в основном сосредоточены шумы. Поэтому (при удачно выбранных  $\alpha$  и  $p$ ) метод регуляризации Тихонова должен давать более точное восстановление дефокусированного изображения, чем метод параметрической фильтрации Винера. В этом мы уже неоднократно убеждались при реконструкции смазанных изображений.

Отметим, что формулы (2.72)–(2.85) должны быть реализованы в дискретном виде с использованием  $m$ -функций `fft2.m` и `ifft2.m` (двухмерное прямое и обратное ДПФ и БПФ) [36, с. 126–128], [45, с. 142, 519].

**Собственные  $m$ -функции для реконструкции дефокусированных изображений.** Разработано несколько собственных  $m$ -функций в системе MatLab для реконструкции дефокусированных (и зашумленных) изображений методом двухмерного ПФ и регуляризации Тихонова согласно (2.84), (2.85), (2.74), (2.75), (2.58)–(2.60), (2.56). Основная функция — это `refocusingT.m`, заголовок и инструкция к которой приведены ниже. Заметим, что если изображение является цветным (RGB), то оно обрабатывается способом векторной обработки (ср. с. 148).

```
function w_alpha = refocusingT(g,r,type,h,alpha,p) %собственная m-функция
%Реконструкция (рефокусирование) дефокусированного изображения
%методом двухмерного ПФ и регуляризации Тихонова.
%Входные параметры (аргументы):
%g - вещественная матрица дефокусированного изображения
%   типа gray или RGB размера mxn класса uint8 или double,
%r - полуширина квадратного окна в пкс (double),
%   в которое должна быть вписана ФПТ (PSF) h,
%type - если type='truncation', то реконструируется
%   дефокусированное изображение с усечением, а если
%   type='diffusion', то с размытием краев,
%h - функция рассеяния точки - ФПТ, PSF (double),
%   которая должна быть оформлена в виде m-функции h(i,j,r)
%   (на основе ее формируется матрица M, см. (2.69)),
%alpha>0 - параметр регуляризации,
%p>=0 - порядок регуляризации.
%Выходной параметр (аргумент):
%w_alpha - вещ. матрица реконструированного изображения
%   такого же типа, как g, класса uint8, размера
%   (m+2r)x(n+2r), если type='truncation', и
%   (m-2r)x(n-2r), если type='diffusion'.
%Пример обращения к refocusingT:
w_alpha = refocusingT(g,r,'truncation',@h_disk,alpha,p);
```

**Примеры реконструкции дефокусированных изображений.** С помощью функции `refocusingT.m` была выполнена реконструкция ряда изображений.

Было реконструировано (рефокусировано) серое *текстовое изображение* (см. вторую часть `RecDefImTt.m` и рис. 2.59, 2.62) с помощью следующих операторов:

```
%RecDefImTt.m (вторая часть).
%Реконструкция (рефокусирование) дефокусированного усеченного
%изображения g1 методом 2-мерного ПФ и регуляризации Тихонова:
w_alpha1 = refocusingT(g1,r,'truncation',@h_disk,1e-2,2); %618x690
sig_w_a1 = rmsd(w_alpha1,w); %=0.4915 (СКО w_alpha1 от w)
figure; %Fig. 2.62:
subplot(2,2,1); imshow(uint8(w_alpha1), []);
title('w_alpha1, \alpha=1e-2, p=2, \sigma=0.4915');
%Реконструкция дефокусированного изображения g2 с размытыми
%краями методом 2-мерного ПФ и регуляризации Тихонова:
w_alpha2 = refocusingT(g2,r,'diffusion',@h_disk,1e-7,2); %618x690
sig_w_a2 = rmsd(w_alpha2,w); %=0.0132 (СКО w_alpha2 от w)
subplot(2,2,2); imshow(uint8(w_alpha2), []);
title('w_alpha2, \alpha=1e-7, p=2, \sigma=0.0132');
%Для сравнения
%реконструкция дефокусированного изображения defoc
%методом параметрической фильтрации Винера (два варианта):
W_K = deconvwnr(defoc,PSF,1e-7); %638x710 (1-й вариант)
[M,N] = size(W_K);
W_Kr = W_K(1+r:M-r,1+r:N-r); %вырезка 618x690 из W_K
sig_W_Kr = rmsd(W_Kr,w); %=0.379(MatLab7.0.1) и 0.0100(MatLab7.10.0)
subplot(2,2,3); imshow(uint8(W_Kr), []);
title('W_Kr, K=1e-7, \sigma=0.0100');
W_Ku = deconvwnr(uint8(defoc),PSF,1e-4); %638x710 (2-й вариант)
W_Kur = W_Ku(1+r:M-r,1+r:N-r); %вырезка 618x690 из W_Ku
sig_W_Kur = rmsd(W_Kur,w); %=0.0906(MatLab7.0.1) и 0.0510(MatLab7.10.0)
subplot(2,2,4); imshow(uint8(W_Kur), []);
title('W_Kur, K=1e-4, \sigma=0.0510');
```

Результаты, приведенные на рис. 2.62, показывают, что реконструкция изображения `g1` с усечением выполнена с большой погрешностью  $\sigma$ , в частности, с темным фоном и с ложными волнами (см. изображение `w_alpha1` на рис. 2.62) в то время, как реконструкция изображения `g2` с размытыми краями выполнена практически без погрешностей (см. изображение `w_alpha2` на рис. 2.62). Что касается метода параметрической фильтрации Винера, то результат сильно зависит от версии `deconvwnr` и от характера обращения к функции `deconvwnr`, а именно, при обращении: `W_K=deconvwnr(defoc,PSF,1e-5)` (1-й вариант) получаемое изображение `W_K` практически не отличается от дефокусированного изображения `defoc` (см. `W_K` на рис. 2.62), а

w\_alpha1,  $\alpha=1e-2$ ,  $p=2$ ,  $\sigma=0.4915$       w\_alpha2,  $\alpha=1e-7$ ,  $p=2$ ,  $\sigma=0.0132$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$$

$$f_1 = 300, \quad f = 3.5,$$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$$f_2 = 3.54, \quad q = f_1/f_2 = 84.7.$$

W\_Kr,  $K=1e-5$ ,  $\sigma=0.379$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$$

$$f_1 = 300, \quad f = 3.5,$$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$$f_2 = 3.54, \quad q = f_1/f_2 = 84.7.$$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$$

$$f_1 = 300, \quad f = 3.5,$$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$$f_2 = 3.54, \quad q = f_1/f_2 = 84.7.$$

W\_Kur,  $K=1e-5$ ,  $\sigma=0.0906$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$$

$$f_1 = 300, \quad f = 3.5,$$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$$f_2 = 3.54, \quad q = f_1/f_2 = 84.7.$$

MatLab 7.0.1

W\_Kr,  $K=1e-7$ ,  $\sigma=0.0100$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$$

$$f_1 = 300, \quad f = 3.5,$$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$$f_2 = 3.54, \quad q = f_1/f_2 = 84.7.$$

W\_Kur,  $K=1e-4$ ,  $\sigma=0.0510$

$$\frac{1}{f_1} + \frac{1}{f_2} = \frac{1}{f}. \quad (2.1)$$

$$f_1 = 300, \quad f = 3.5,$$

$$f_2 = \left( \frac{1}{f} - \frac{1}{f_1} \right)^{-1}, \quad (2.4)$$

$$f_2 = 3.54, \quad q = f_1/f_2 = 84.7.$$

MatLab 7.10.0

Рис. 2.62. Реконструкция дефокусированного серого изображения (см. рис. 2.59):

w\_alpha1 — реконструкция усеченного изображения g1 методом

2-мерного ПФ и регуляризации Тихонова,

w\_alpha2 — реконструкция изображения g2 с размытием краев методом

2-мерного ПФ и регуляризации Тихонова,

W\_K — реконструкция изображения defoc методом параметрической  
фильтрации Винера:  $W_K = \text{deconvwnr}(\text{defoc}, \text{PSF}, 1e-7)$ ;

W\_Ku — реконструкция изображения defoc методом параметрической  
фильтрации Винера:  $W_Ku = \text{deconvwnr}(\text{uint8}(\text{defoc}), \text{PSF}, 1e-4)$ ;

(приведены вырезки W\_Kr и W\_Kur, полученные в разных версиях системы MatLab).

Значения параметров  $\alpha$  и  $K$  были сначала подобраны грубо путем визуальной оценки реконструированных изображений  $w_\alpha$  и  $w_K$ , а затем более точно по условию минимума  $\sigma_{\text{rel}}(\alpha)$  и  $\sigma_{\text{rel}}(K)$  (см. (2.41) и (2.44)).

Размер каждого изображения  $618 \times 690$

при обращении: `W_Ku=deconvwnr(uint8(defoc),PSF,1e-5)` (2-й вариант) реконструкция значительно лучше (см. `W_Ku` на рис. 2.62), хотя имеют место ложные волны и погрешность  $\sigma$  заметно больше, чем у изображения `w_alpha2`. Сказанное относится к `deconvwnr`, приведенной в подверсии MatLab 7.0.1. Результат же, полученный по `deconvwnr`, приведенной в подверсии MatLab 7.10.0, заметно лучше, хотя и уступает `w_alpha2`.

Результаты, приведенные на рис. 2.62, говорят о том, что использование приема "размытие краев изображения" может заметно повысить точность реконструкции. Кроме того, эти результаты подтверждают то, что метод регуляризации Тихонова (см. (2.84), (2.85)) в принципе точнее метода параметрической фильтрации Винера (см. (2.80), (2.81)).

Было реконструировано также *цветное (RGB) изображение* – см. `RecDefImTm1.m` и рис. 2.60, 2.61, 2.63, 2.65, 2.66, а также `RecDefImTm2.m` и рис. 2.60, 2.61, 2.64, 2.65, 2.67. Реконструкция была выполнена в отсутствие шума и при его наличии.

В головной программе `RecDefImTm1.m` после выполнения дефокусирования (см. рис. 2.60) добавляется импульсный цветной (RGB) шум с помощью внешней `m`-функции `imnoise`. Затем решается обратная задача. Сначала выполняется фильтрация цветного шума. При этом поскольку внешняя `m`-функция `medfilt2` (медианная фильтрация), а также `wiener2` (адаптивная винеровская фильтрация) не фильтрует цветные шумы, была разработана собственная `m`-функция `medfilt2c` (а также `wiener2c`) для фильтрации цветных (RGB) шумов. Ниже приведена `m`-функция `medfilt2c.m`<sup>14</sup>

```
function f = medfilt2c(g,nhood)
%Фильтрация gray- или RGB-шума на gray- или RGB-изображении
%методом медианной фильтрации. Рекомендация: функция
%medfilt2c наиболее эффективно фильтрует импульсный шум.
%Если же шум гауссовый, то рекомендуется использовать
%функцию wiener2c.m - адаптивная винеровская фильтрация шума.
%Входные параметры (аргументы):
%g - исходное зашумленное gray- или RGB-изображение,
%nhood - скользящее окно (neighborhood) размера mxn, в пределах
% которого интенсивности выстраиваются в порядке возрастания
% и средней точке окна присваивается интенсивность под номером
% mxn/2 (медианный отсчет). При этом мощные и слабые шумы
% (это, как правило, импульсные шумы) отсеиваются.
%Выходной параметр (аргумент):
%f - обработанное изображение такого же типа, класса и размера,
% как g.
%Отличие собственной функции medfilt2c от внешней функции
%medfilt2 состоит в том, что medfilt2 обрабатывает лишь gray-
%изображения, а medfilt2c - как gray-, так и RGB-изображения.
%Пример обращения к medfilt2c:
```

<sup>14</sup>Текст `wiener2c.m` см. на CD.

```

%gf = medfilt2c(gn,[5 5]); %фильтрация цв. шума на изображении gn
%Литература:
%Дьяконов В., Абраменкова И. MATLAB. Обработка сигналов и
% изображений. СПб.: Питер, 2002, с. 550.
kg = ndims(g); %если kg=2, то g - gray, а если kg=3, то g - RGB
if (kg==2) %gray
    f = medfilt2(g,nhood);
else %RGB
    f(:,:,1) = medfilt2(g(:,:,1),nhood);
    f(:,:,2) = medfilt2(g(:,:,2),nhood);
    f(:,:,3) = medfilt2(g(:,:,3),nhood);
end

```

С помощью функции `medfilt2c.m` была выполнена фильтрация цветного шума. После этого при различных значениях  $\alpha$  и  $K$  была выполнена реконструкция изображений методами регуляризации Тихонова и параметрической фильтрации Винера (`refocusingT.m` и `deconvwnr.m`) и рассчитаны зависимости  $\sigma_{rel}(\alpha)$  и  $\sigma_{rel}(K)$  (см. рис. 2.63 и 2.66).

Ниже приведена вторая часть `RecDefImTm1.m`.

```

%RecDefImTm1.m (вторая часть)
%Фильтрация импульсного RGB-шума с помощью m-функции medfilt2c.m:
g1f = medfilt2c(g1n,[3 3]); %фильтрация изображения с усечением
g2f = medfilt2c(g2n,[3 3]); %фильтрация изображения с размытием
defocsf = medfilt2c(defocsn,[3 3]); %фильтрация изображения defocsn
defoccf = medfilt2c(defocsn,[3 3]); %фильтрация изображения defocsn
figure; %Fig. 2.65:
subplot(2,2,1), imshow(uint8(g1f)), title('g1f 208x238x3')
subplot(2,2,2), imshow(uint8(g2f)), title('g2f 228x258x3')
subplot(2,2,3), imshow(uint8(defocsf)), title('defocsf 218x248x3')
subplot(2,2,4), imshow(uint8(defoccf)), title('defoccf 218x248x3')
%Реконструкция изображений методом 2-мерного ПФ и рег. Тихонова
%с расчетом кривых sigma_rel(alpha), а также методом параметри-
%ческой фильтрации Винера с расчетом кривых sigma_rel(K):
alpha1=1e-8; i=0;
I=round((log10(1e4)-log10(1e-8))/log10(10^0.5)+1); %I=25
alpha=zeros(1,I); %Выделение памяти
sig_w_a1=zeros(1,I); sig_w_a2=zeros(1,I); %Выделение памяти
sig_W_Ks=zeros(1,I); sig_W_Kc=zeros(1,I); %Выделение памяти
while (alpha1<=1e4+1e-4)
i=i+1; alpha(i)=alpha1;
%Метод ПФ и рег. Тихонова с усечением:
w_alpha1 = refocusingT(g1,r,'truncation',@h_disk,alpha1,1); %без шума
%или (после фильтрации шума):

```

```

w_alpha1 = refocusingT(g1f,r,'truncation',@h_disk,alpha1,1);
sig_w_a1(i) = rmsd(w_alpha1,w); %СКО w_alpha1 от w
%Метод ПФ и рег. Тихонова с размытием:
w_alpha2 = refocusingT(g2,r,'diffusion',@h_disk,alpha1,1); %без шума
%или (после фильтрации шума):
w_alpha2 = refocusingT(g2f,r,'diffusion',@h_disk,alpha1,1);
sig_w_a2(i) = rmsd(w_alpha2,w); %СКО w_alpha2 от w
%Метод параметрической фильтрации Винера:
K = alpha1;
W_Ks = deconvwnr(uint8(defocs),PSF,K); %без шума
%или (после фильтрации шума):
W_Ks = deconvwnr(uint8(defocsf),PSF,K);
sig_W_Ks(i) = rmsd(W_Ks,w);
W_Kc = deconvwnr(uint8(defoccc),PSF,K); %без шума
%или (после фильтрации шума):
W_Kc = deconvwnr(uint8(defocccf),PSF,K);
sig_W_Kc(i) = rmsd(W_Kc,w);
fprintf('\n lg(alpha1)=%5.1f',log10(alpha1));
alpha1=alpha1*10^0.5;
end %while (alpha)
fprintf('\n ');
%2D-графика:
lga=log10(alpha); %массив
alpha1=1e-8; i=0;
I1=round((log10(1e4)-log10(1e-8))/log10(10^0.1)+1); %I1=121
alphas=zeros(1,I1);
while (alpha1<=1e4+1e-4)
    i=i+1; alphas(i)=alpha1; %s означает small step
    alpha1=alpha1*10^0.1;
end %while (alpha)
lgas=log10(alphas); %массив (более частая сетка)
%Для ускорения используется интерполяция многочленами Эрмита:
figure; %Fig. 2.63 или 2.66:
plot(lgas,interp1(lga,sig_w_a1,lgas,'cubic'),'-k',...
     lgas,interp1(lga,sig_w_a2,lgas,'cubic'),'-k',...
     lgas,interp1(lga,sig_W_Ks,lgas,'cubic'),'-k',...
     lgas,interp1(lga,sig_W_Kc,lgas,'cubic'),'-k',...
     'LineWidth',2.5);
grid on; axis([-8,4,0,1]);
title('\sigma_{rel}(\alpha), \sigma_{rel}(\{it{K}\})','FontSize',16);
text(2.5,-0.065,'lg \alpha, lg\it K','FontSize',16);
text(0.5,0.35,'\sigma(w1_\alpha)','FontSize',16);
text(0.7,0.15,'\sigma(w2_\alpha)','FontSize',16);

```

```
text(-3.1,0.25,'\sigma(W\_Ks)', 'FontSize',16);
text(-5.1,0.17,'\sigma(W\_Kc)', 'FontSize',16);
```

На рис. 2.63 приведены зависимости относительных погрешностей  $\sigma_{\text{rel}}(\alpha)$  и  $\sigma_{\text{rel}}(K)$  реконструкции дефокусированного цветного незашумленного изображения.

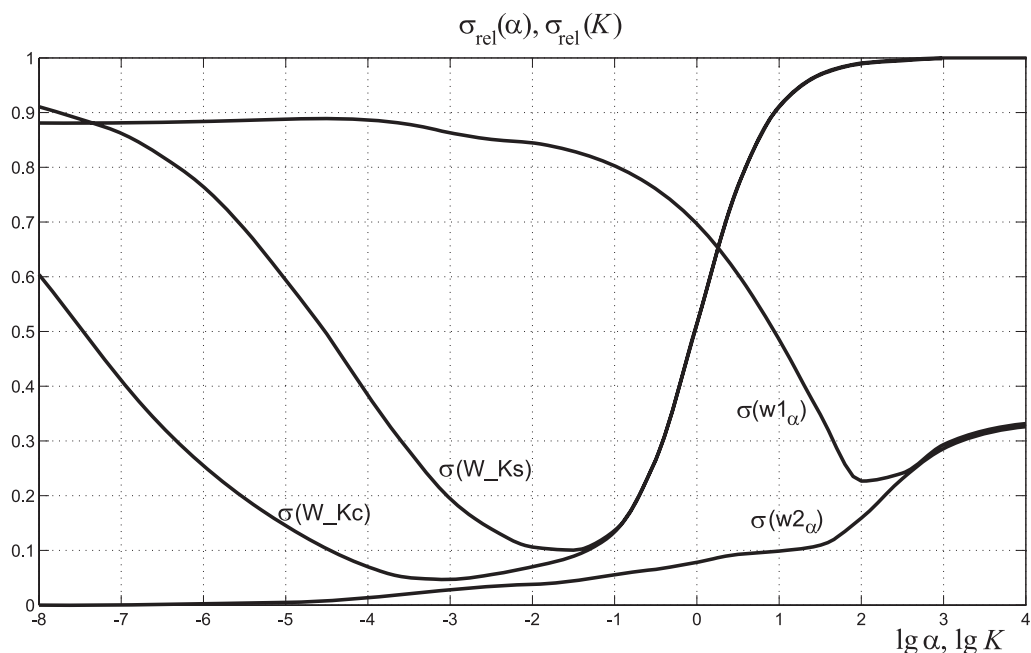


Рис. 2.63. Зависимость относительной погрешности  $\sigma_{\text{rel}}$  реконструкции дефокусированного цветного незашумленного изображения от  $\alpha$  и  $K$ :  $\sigma(w1_\alpha)$ ,  $\sigma(w2_\alpha)$ ,  $\sigma(W\_Ks)$  и  $\sigma(W\_Kc)$  — погрешности реконструкции изображений  $g1$ ,  $g2$ ,  $\text{defocs}$  и  $\text{defocsc}$  (см. рис. 2.60)

Далее приведена вторая часть головной программы `RecDefImTm2.m` реконструирования дефокусированных цветных (RGB) изображений, незашумленных (см. рис. 2.60) или зашумленных (см. рис. 2.61), но очищенных от шума с помощью `medfilt2c.m` (см. далее рис. 2.65).<sup>15</sup> Реконструкция выполнена методом 2-мерного ПФ и регуляризации Тихонова (с усечением и размытием) и методом параметрической фильтрации Винера.<sup>16</sup>

```
%RecDefImTm2.m (вторая часть)
%Обратная задача - реконструкция дефокусированного изображения.
%Фильтрация импульсного RGB-шума с помощью m-функции medfilt2c.m
%и рефокусирование дефокусированного усеченного изображения
```

<sup>15</sup> Шум импульсный.

<sup>16</sup> На CD см. аналогичную программу `RecDefImTw2.m`, в которой шум является гауссовым и изображение очищается от него с помощью функции `wiener2c.m`.

```

%g1n методом 2-мерного ПФ и регуляризации Тихонова:
g1f = medfilt2c(g1n,[3 3]); %фильтрация цветного шума
%w_alpha1 = refocusingT(g1,r,'truncation',@h_disk,1e2,1); %рег.Тих.
w_alpha1 = refocusingT(g1f,r,'truncation',@h_disk,1e2,1); %рег.Тих.
sig_w_a1 = rmsd(w_alpha1,w); %=0.2266 (СКО w_alpha1 от w)
%Фильтрация шума и рефокусирование дефокусированного изображения
%g2n с размытыми краями методом 2-мерного ПФ и рег. Тихонова:
g2f = medfilt2c(g2n,[3 3]); %фильтрация цветного шума
%w_alpha2 = refocusingT(g2,r,'diffusion',@h_disk,10^(-7),1); %рег.Тих.
w_alpha2 = refocusingT(g2f,r,'diffusion',@h_disk,10^1.2,1); %рег.Тих.
sig_w_a2 = rmsd(w_alpha2,w); %3e-5, 0.1067 (СКО w_alpha2 от w)
%Для сравнения
%фильтрация шума и рефокусирование дефокусированных изображений
%defocsn и defocsn методом параметрической фильтрации Винера:
defocsf = medfilt2c(defocsn,[3 3]); %фильтрация шума
%W_Ks = deconvwnr(uint8(defocsf),PSF,10^(-1.6)); %пар.фильтр.Винера
W_Ks = deconvwnr(uint8(defocsf),PSF,10^(-1.6)); %пар.фильтр.Винера
sig_W_Ks = rmsd(W_Ks,w); %=0.0991, 0.1009(MatLab7.10.0)
defoccf = medfilt2c(defocsn,[3 3]); %фильтрация шума
%W_Kc = deconvwnr(uint8(defoccf),PSF,10^(-3.1)); %пар.фильтр.Винера
W_Kc = deconvwnr(uint8(defoccf),PSF,10^(-2.4)); %пар.фильтр.Винера
sig_W_Kc = rmsd(W_Kc,w); %=0.0475, 0.0739(MatLab7.10.0)
figure; %Fig. 2.65
subplot(2,2,1), imshow(uint8(g1f)), title('g1f 208x238x3')
subplot(2,2,2), imshow(uint8(g2f)), title('g2f 228x258x3')
subplot(2,2,3), imshow(uint8(defocsf)), title('defocsf 218x248x3')
subplot(2,2,4), imshow(uint8(defoccf)), title('defoccf 218x248x3')
figure; %Fig. 2.64 или 2.67
subplot(2,2,1); imshow(uint8(w_alpha1), []);
title('w_alpha1, \alpha=10^2, p=1, \sigma=0.2266');
subplot(2,2,2); imshow(uint8(w_alpha2), []);
%title('w_alpha2, \alpha=10^{-7}, p=1, \sigma=3e-5');
title('w_alpha2, \alpha=10^{1.2}, p=1, \sigma=0.1067');
subplot(2,2,3); imshow(uint8(W_Ks), []);
%title('W_Ks, K=10^{-1.6}, \sigma=0.0991');
title('W_Ks, K=10^{-1.6}, \sigma=0.1009');
subplot(2,2,4); imshow(uint8(W_Kc), []);
%title('W_Kc, K=10^{-3.1}, \sigma=0.0475');
title('W_Kc, K=10^{-2.4}, \sigma=0.0739');

```

На рис. 2.64 приведены реконструкции дефокусированных незашумленных цветных (RGB) изображений, представленных на рис. 2.60, методом 2-мерного ПФ с регуляризацией Тихонова (с усечением и с размытием) и методом параметрической фильтрации Винера (с двумя типами "граничных условий") при  $\alpha = \alpha_{\text{opt}}$  и  $K = K_{\text{opt}}$ .





Рис. 2.64. Реконструкция дефокусированного изображения без шума (см. рис. 2.60):

- $w\_alpha1$  — реконструкция усеченного изображения  $g1$  методом 2-мерного ПФ и регуляризации Тихонова,
  - $w\_alpha2$  — реконструкция изображения  $g2$  с размытыми краями методом 2-мерного ПФ и регуляризации Тихонова,
  - $W\_Ks$  — реконструкция изображения  $defocs$  методом параметрической фильтрации Винера,
  - $W\_Kc$  — реконструкция изображения  $defocs$  методом параметрической фильтрации Винера.
- Размер каждого изображения  $218 \times 248 \times 3$

Из рис. 2.63 и 2.64 видно, что в случае, когда (цветное) изображение лишь дефокусировано, но не зашумлено (и ФРТ известна точно), наилучшую (практически точную) реконструкцию дает метод 2-мерного ПФ с тихоновской регуляризацией и с размытием краев изображения.

На рис. 2.61, 2.65–2.67 представлены результаты, когда (цветное) изображение не только дефокусировано, но и зашумлено (импульсным шумом) (ФРТ известна точно).

Видим, что, во-первых, предварительная фильтрация импульсного шума функцией `medfilt2c.m` заметно очистила от шума изображения (см. рис. 2.65), а, во-вторых, последующее устранение дефокусирования методами 2-мерного ПФ с регуляризацией Тихонова с размытием краев и параметрической фильтрации Винера с различными вариантами "граничных условий" дает примерно одинаковый результат: метод Тихонова дает большую  $\sigma_{rel}$ , чем метод Винера (см. рис. 2.66 и 2.67), но метод Тихонова с размытием дает меньший эффект Гиббса, чем метод Винера (см. рис. 2.67).



Рис. 2.65. Результат фильтрации импульсного шума на изображениях  $g_1$ ,  $g_2$ ,  $defocs$  и  $defoccf$  (см. рис. 2.60) с помощью функции `medfilt2c.m`

**Метод итераций с квадратурами.** Обычно интегральное уравнение (2.56) решается методом двумерного преобразования Фурье с использованием некоторого устойчивого метода (фильтрации Винера, регуляризации Тихонова и т.д.). Применение преобразования Фурье позволяет существенно снизить требуемую компьютерную память (аналогично тому, как при решении одномерного уравнения (2.13) методом ПФ требуются только векторы и не требуются матрицы). Однако преобразование Фурье описывает неадекватно физическую природу процесса дефокусирования изображения, поскольку при дефокусировании реальных (не модельных) изображений сама природа использует лишь операцию суммирования (интегрирования, накопления, усреднения) — см. (2.54), (2.62)–(2.65), (2.67), (2.68), (2.70), (2.71). Кроме того, преобразование Фурье при численном решении реализуется в виде дискретного преобразования Фурье (ДПФ), а ДПФ обладает такими "отрицательными" свойствами, как периодичность, наложение, возможная перестановка относительно середины основного интервала, эффект Гиббса и др., что искажает искомое решение  $w(\xi, \eta)$ .

Указанные недостатки метода ПФ уже отмечались в п. 2.3 и 2.4, в задаче реконструкции смазанных изображений, описываемой набором одномерных интегральных уравнений Фредгольма I рода типа свертки. Их решение методом ПФ с регуляризацией Тихонова показало, что более адекватным (и точным) является решение методом квадратур (также с регуляризацией Тихонова). Реконструкция ряда смазанных изоб-

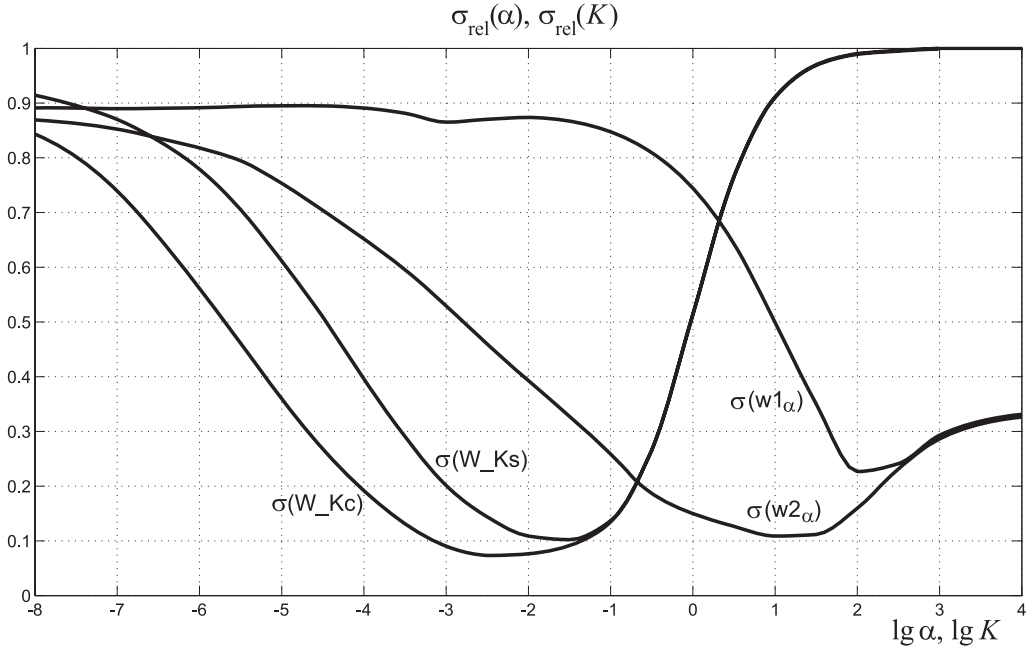


Рис. 2.66. Зависимость относительной погрешности  $\sigma_{\text{rel}}$  реконструкции дефокусированного цветного зашумленного изображения от  $\alpha$  и  $K$ :  $\sigma(w1_\alpha)$ ,  $\sigma(w2_\alpha)$ ,  $\sigma(W\_Ks)$  и  $\sigma(W\_Kc)$  — погрешности реконструкции изображений `g1f`, `g2f`, `defocsf` и `defoccf` (см. рис. 2.65)

ражений подтвердила это.

Возникает идея применить и для решения двухмерного уравнения (2.56) (а также (2.61)) метод квадратур (точнее, кубатур) с регуляризацией Тихонова. Проанализируем эту идею. Запишем (2.56) с использованием конечных пределов интегрирования, полагая, что размеры  $w$  и  $g$ , вообще говоря, разные (см. рис. 2.56–2.58).

$$\int_a^b \int_c^d h(x - \xi, y - \eta) w(\xi, \eta) d\xi d\eta = g(x, y), \quad a' \leq x \leq b', \quad c' \leq y \leq d'. \quad (2.86)$$

Если каждый интеграл в (2.86) заменить конечной суммой по некоторой квадратурной формуле (например, трапеций), то получим СЛАУ относительно значений  $w$  в узлах дискретных сеток. При этом матрица СЛАУ  $h$  будет 4-мерной, а  $w$  и  $g$  — 2-мерными матрицами. Если такую СЛАУ привести к СЛАУ с 2-мерной матрицей  $h$  (и 1-мерными  $w$  и  $g$ ), то матрица  $h$  получится очень большого размера, а именно,  $m^2 \times n^2$ , где  $m$  — число узлов дискретизации по  $x$  и  $\xi$ , а  $n$  — число узлов по  $y$  и  $\eta$ .

Если, например, изображение есть матрица  $400 \times 400$  пикселей, т.е.  $m = n = 400$ , то получим систему  $mn = 160000$  уравнений с  $mn = 160000$  неизвестными и с матрицей  $m^2 \times n^2 = 160000 \times 160000$  (ср. [35, с. 396]). Ясно, что возникает труднейшая проблема решения такой гигантской СЛАУ даже на суперкомпьютерах и даже с учетом того,

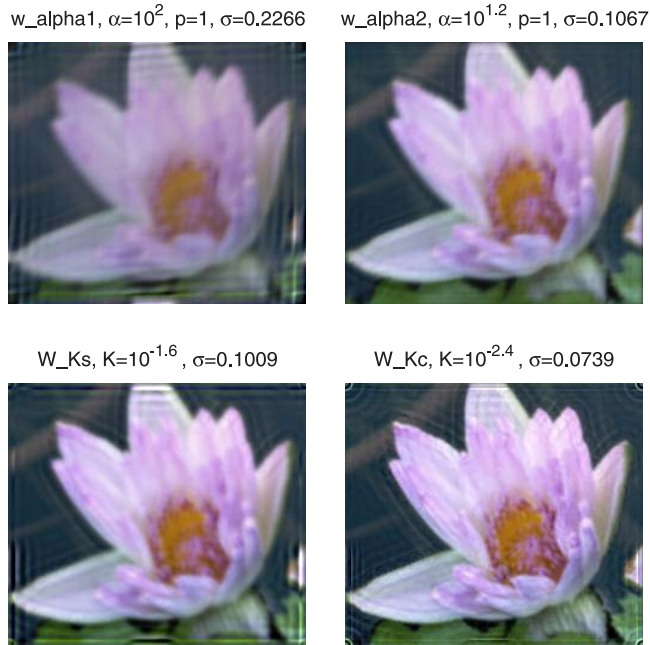


Рис. 2.67. Реконструкция дефокусированного цветного зашумленного изображения (см. рис. 2.61 и 2.65):  
**w\_alpha1** — реконструкция усеченного изображения **g1f** методом 2-мерного ПФ и регуляризации Тихонова,  
**w\_alpha2** — реконструкция изображения **g2f** с размытыми краями методом 2-мерного ПФ и регуляризации Тихонова,  
**W\_Ks** — реконструкция изображения **defocsf** методом параметрической фильтрации Винера,  
**W\_Kc** — реконструкция изображения **defoccf** методом параметрической фильтрации Винера.  
 Размер каждого изображения  $218 \times 248 \times 3$

что матрица  $h$  является теплицевой (разностной). Тем не менее, существуют матричные, или алгебраические методы — это методы, использующие квадратуры, например, итеративный метод (сопряженных градиентов — СГ и др.) минимизации сглаживающего функционала Тихонова [164, 167]. Мы же остановимся на *методе итеративной регуляризации Фридмана* [22, с. 272], [114, с. 79], [130].

Для улучшения свойств уравнения (2.86) положим размеры изображений  $w$  и  $g$  одинаковыми, т.е.  $a = a'$ ,  $b = b'$ ,  $c = c'$ ,  $d = d'$ . Тогда

$$Aw \equiv \int_a^b \int_c^d h(x - \xi, y - \eta) w(\xi, \eta) d\xi d\eta = g(x, y), \quad a \leq x \leq b, \quad c \leq y \leq d, \quad (2.87)$$

где  $A$  — интегральный оператор. Кроме того, полагаем, что ядро  $h$  является симметричным, т.е.  $h(x, y) = h(|x|, |y|)$ . В таком случае оператор  $A$  становится симметричным, положительно определенным. Это позволяет записать *схему итераций* (последовательных приближений) *Фридмана* в простейшей форме [21, с. 22], [22, с. 272]:

$$w_k(x, y) = w_{k-1}(x, y) + \nu \left[ g(x, y) - \iint_a^b \iint_c^d h(x - \xi, y - \eta) w_{k-1}(\xi, \eta) d\xi d\eta \right],$$

$$k = 1, 2, 3, \dots, \quad a \leq x \leq b, \quad c \leq y \leq d, \quad (2.88)$$

$$0 < \nu < 2/\|A\|, \quad (2.89)$$

где  $k$  — номер итерации, причем  $w_0(x, y)$  — некоторое начальное приближение, например,  $w_0(x, y) = 0$  или  $w_0(x, y) = \nu g(x, y)$  или  $w_0(x, y) = g(x, y)$ ,  $\|A\|$  — норма оператора, в качестве которой можно взять норму Гильберта–Шмидта [22, с. 273]

$$\|A\| = \left\{ \iint_a^b \iint_c^d h^2(x, y) dx dy \right\}^{1/2}. \quad (2.90)$$

Обычно  $h(x, y)$  записывают в виде (2.58), (2.59) и т.п. В этом случае  $\|A\| = 1$  и в результате

$$0 < \nu < 2. \quad (2.91)$$

Численная (дискретная) реализация схемы (2.88) примет вид (см. рис. 2.68):

$$w_k(j, i) = w_{k-1}(j, i) + \nu \left[ g(j, i) - \frac{1}{Q} \sum_{j'=\max\{1, j-r\}}^{\min\{m, j+r\}} \sum_{i'=\max\{1, i-r\}}^{\min\{n, i+r\}} h(i - i', j - j') w_{k-1}(j', i') \right],$$

$$(2.92)$$

$$k = 1, 2, 3, \dots, K, \quad j = 1, 2, \dots, m, \quad i = 1, 2, \dots, n, \quad 0 < \nu < 2,$$

где  $K$  — максимальное число итераций. Формула (2.92) справедлива для случая, когда ФРТ  $h$  — любого вида, но разностная и заключенная в пределах квадратного окна  $2r \times 2r$  (рис. 2.54), а вне окна равная нулю. Заметим, что двойная сумма в (2.92) весьма напоминает двойную сумму в (2.65), однако использование функций  $\min$  и  $\max$  в (2.92) устраняет возможность выхода индексов  $j'$  и  $i'$  за пределы  $[1, m]$  и  $[1, n]$  соответственно и, кроме того, повышает скорость вычисления двойной суммы в (2.92).

В схеме (2.92) благодаря наличию нормирующего делителя  $Q$  в качестве ФРТ  $h$  можно брать выражения (2.58'), (2.59') и т.п.

Однако, как уже отмечалось выше, MatLab наиболее эффективен, когда он оперирует исключительно с матрицами, а в схеме (2.92) фигурируют не только матрицы

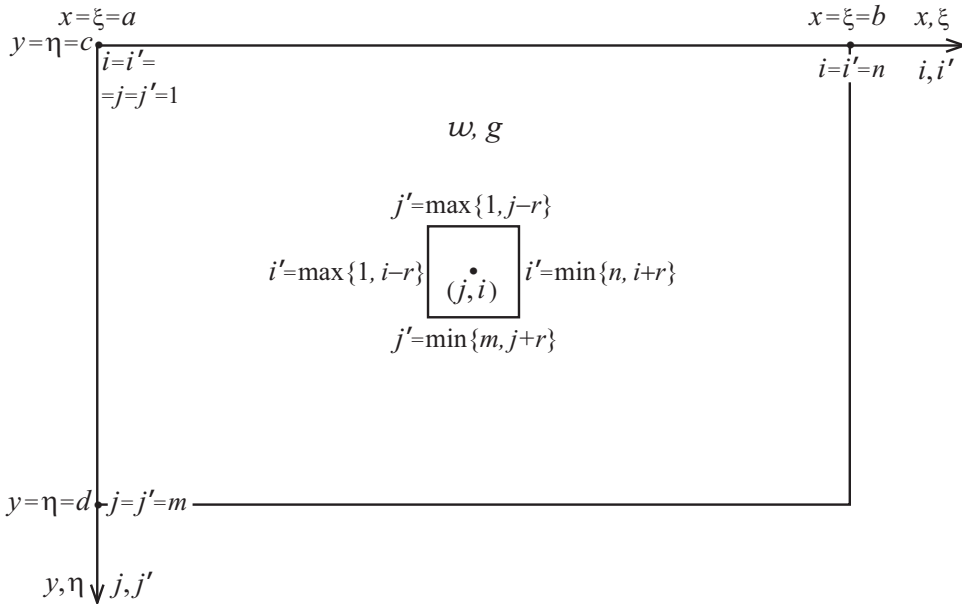


Рис. 2.68. Обозначения для схемы (2.92)

$w$  и  $g$ , но и функция  $h$ , что замедляет вычисления. Поэтому, как и выше, введем  $M$  — матрицу, связанную с  $h$  соотношением (2.69). Тогда (2.92) запишется в виде:

$$w_k(j, i) = w_{k-1}(j, i) + \nu \left[ g(j, i) - \frac{1}{Q} \sum_{j'=\max\{1, j-r\}}^{\min\{m, j+r\}} \sum_{i'=\max\{1, i-r\}}^{\min\{n, i+r\}} M(j-j'+r+1, i-i'+r+1) w_{k-1}(j', i') \right], \quad (2.93)$$

$$k = 1, 2, 3, \dots, K, \quad j = 1, 2, \dots, m, \quad i = 1, 2, \dots, n, \quad 0 < \nu < 2.$$

Численные эксперименты на компьютере показали, что вычисления на MatLab по схеме (2.93) выполняются на 2 порядка (!) быстрее, чем по схеме (2.92).

Разработана собственная m-функция `refocusingF.m` для реконструкции (рефокусирования) дефокусированных (а также зашумленных, серых и цветных) изображений методом итераций Фридмана согласно (2.93). Ниже приведен заголовок этой функции и инструкция к ней.

```
function [w_k_opt, sigma_rel] = refocusingF(g, w0, r, h, nu, k_opt, K, w)
%Реконструкция (рефокусирование) дефокусированного изображения
%методом итеративной регуляризации Фридмана.
%Входные параметры (аргументы):
%g - вещественная матрица дефокусированного изображения
% типа gray или RGB размера mхn класса uint8 или double,
%w0 - начальное приближение - матрица такого же типа,
```

```

% размера и класса, что и g,
%r - полуширина квадратного окна в пкс (double),
% в которое должна быть вписана ФРТ (PSF) h,
%h - функция рассеяния точки - ФРТ, PSF (double),
% которая должна быть оформлена в виде m-функции h(i,j,r)
% (на основе ее формируется матрица M, см. (2.69)),
%nu - параметр метода Фридмана в диапазоне: 0<nu<2,
%k_opt<=K - некоторый (оптимальный или квазиоптимальный) номер итерации,
%K>=1 - максимальное число итераций (последовательных приближений)
% (если nargout=1, то K=k_opt, а если nargout=2, то K>=k_opt),
%w - точное изображение - матрица такого же типа, размера
% и класса, что и g (используется при реконструкции лишь
% модельного изображения).
%Выходные параметры (аргументы):
%w_k_opt - вещ. матрица реконструированного изображения в k_opt-й
% итерации такого же типа и размера, как g, класса double,
%sigma_rel - относительное СКО (см. (2.94)) в функции k - вектор
% длиной K класса double (вычисляется при реконструкции
% лишь модельного изображения).
%Возможные варианты обращений к refocusingF:
%w5 = refocusingF(g,w0,10,@h_disk,0.3,5);
%или
%[w5,sig15] = refocusingF(g,w0,10,@h_disk,0.3,5,15,w);

```

С помощью m-функции `refocusingF.m` были реконструированы (рефокусированы) некоторые дефокусированные изображения методом итеративной регуляризации Фридмана. При реализации данного метода важными являются вопросы выбора значения параметра  $\nu$ , максимального числа итераций  $K$  (играющего роль параметра регуляризации) и начального приближения  $w_0$ .

Было рефокусировано *серое портретное изображение* (см. `RecDefImFg.m` и рис. 2.69–2.71) с помощью следующих операторов:

```

%RecDefImFg.m
%Исходное серое (gray) изображение:
wu = imread('cameraman.tif'); %256x256 uint8
w = double(wu); %256x256 double
r = 5; %квадратное окно имеет размеры 2rx2r
%Дефокусирование с усечением изображения:
g1 = defocusing(w,r,'truncation',@h_gauss); %236x236 double
figure; %Fig. 2.69:
subplot(2,2,1), imshow(uint8(g1)), title('g1 246x246')
%Дефокусирование с размывом краев изображения:
g2 = defocusing(w,r,'diffusion',@h_gauss); %266x266 double
%Обратная задача - реконструкция (рефокусирование)

```



Рис. 2.69. Серое изображение `cameraman.tif`,  
 $g1$  — дефокусированное изображение с усечением согласно (2.70),  
 $g2r$  — дефокусированное изображение с размытием краев  
 согласно (2.71) и с вырезкой до размера  $256 \times 256$ ,  
 $w0 = \nu \cdot g2r$  — начальное приближение  $w_0 = \nu \cdot g2r$ ,  
 $w0 = 0.7w$  — начальное приближение  $w_0 = 0.7\bar{w}$

%дефокусированного изображения методом итераций Фридмана:

```
mg2 = size(g2,1); %266 - кол-во строк
ng2 = size(g2,2); %266 - кол-во столбцов
g2r = g2(1+r:mg2-r,1+r:ng2-r); %вырезка 256x256 из g2
nu = 1.5;
subplot(2,2,2), imshow(uint8(g2r)), title('g2r 256x256')
subplot(2,2,3), imshow(uint8(nu*g2r)), title('w0=\nu*g2r 256x256')
subplot(2,2,4), imshow(uint8(0.7*w)), title('w0=0.7w 256x256')
K = 50;
[w1,sig1] = refocusingF(g2r,0*g2r,r,@h_gauss,nu,27,K,w); %256x256
[w2,sig2] = refocusingF(g2r,nu*g2r,r,@h_gauss,nu,27,K,w); %256x256
[w3,sig3] = refocusingF(g2r,g2r,r,@h_gauss,nu,28,K,w); %256x256
[w4,sig4] = refocusingF(g2r,0.7*w,r,@h_gauss,nu,27,K,w); %256x256
%Графики:
k = 0:K;
figure; %Fig. 2.70:
plot(k,sig1,'-k',k,sig2,'-k',k,sig3,'-k',k,sig4,'-k','LineWidth',2.5);
```



```

sig1_k_opt = sig1(27+1); %=0.1108
figure; %Fig. 2.71:
subplot(2,2,1), imshow(uint8(w1))
title('w1, w0=0, K=27, \sigma_{rel}=0.1108')
sig2_k_opt = sig2(27+1); %=0.1108
subplot(2,2,2), imshow(uint8(w2))
title('w2, w0=\nu*g2r, K=27, \sigma_{rel}=0.1108')
sig3_k_opt = sig3(28+1); %=0.1108
subplot(2,2,3), imshow(uint8(w3))
title('w3, w0=g2r, K=28, \sigma_{rel}=0.1108')
sig4_k_opt = sig4(27+1); %=0.0332
subplot(2,2,4), imshow(uint8(w4))
title('w4, w0=0.7w, K=27, \sigma_{rel}=0.0332')

```

На рис. 2.69 представлены:  $g1$  — дефокусированное изображение с усечением,  $g2r$  — дефокусированное изображение с размытием краев и с вырезкой (чтобы получить размер  $256 \times 256$ ),  $w0$  — начальное приближение  $w_0 = \nu \cdot g2r$  и  $w0 = 0.7w$  — начальное приближение  $w_0 = 0.7\bar{w}$ .

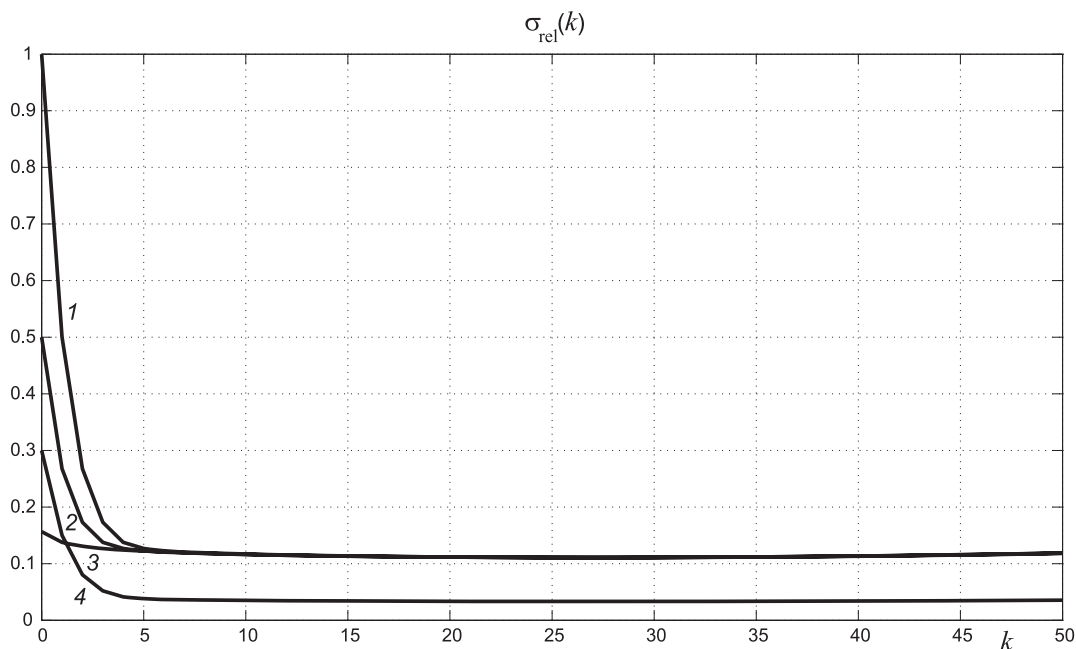


Рис. 2.70. Зависимость погрешности реконструкции дефокусированного изображения методом итераций Фридмана от номера итерации  $k$ :

- 1 — начальное приближение  $w_0 = 0$ ,
- 2 — начальное приближение  $w_0 = \nu \cdot g2r$ ,
- 3 — начальное приближение  $w_0 = g2r$ ,
- 4 — начальное приближение  $w_0 = 0.7\bar{w}$

На рис. 2.70 представлены зависимости относительного среднеквадратического отклонения (СКО) изображения  $w_k$  от точного изображения  $\bar{w}$ , или относительной среднеквадратической погрешности реконструкции изображения в зависимости от номера итерации  $k$  (ср. (2.41), (2.44)):

$$\sigma_{\text{rel}}(k) = \frac{\sqrt{\sum_{j=1}^m \sum_{i=1}^n [(w_k)_{ji} - \bar{w}_{ji}]^2}}{\sqrt{\sum_{j=1}^m \sum_{i=1}^n \bar{w}_{ji}^2}}. \quad (2.94)$$

Из анализа рис. 2.70 можно видеть, что, во-первых, погрешность  $\sigma_{\text{rel}}(k)$  при некотором  $k \approx 10 - 40$  имеет (пологий) минимум, а затем медленно нарастает с ростом  $k$ , т.е. процесс итераций в принципе расходится — проявление некорректности задачи решения уравнения (2.87). Практика решения других задач показывает [22], что в отсутствие шумов обычно достаточно порядка 10 итераций. При наличии же шумов минимум функции  $\sigma_{\text{rel}}(k)$  становится более узким и менее глубоким, а процесс расходимости более значительным (ср. [22, с. 281, рис. 21]).

Во-вторых, процесс итераций, а именно, ход кривой  $\sigma_{\text{rel}}(k)$  зависит не только от значения  $\nu$  и уровня шума, но и от начального приближения  $w_0$ . Из рис. 2.70 и 2.71 видно, что при начальном приближении, весьма близком к  $\bar{w}$  (в нашем случае при  $w_0 = 0.7\bar{w}$ ), можно получить достаточно хорошее восстановление изображения, но такое начальное приближение на практике трудно подобрать.

Было обработано (дефокусировано и рефокусировано) также *цветное (RGB) изображение*:

```
%RecDefImFc.m
%Исходное цветное (RGB) изображение:
RGB = imread('football.jpg'); %256x320x3 uint8
w = double(RGB); %256x320x3 double
%Прямая задача - дефокусирование изображения:
r = 10; %квадратное окно имеет размеры 2rx2r
g1 = defocusing(w,r,'truncation',@h_disk); %236x300x3 double
figure; %Fig. 2.72:
subplot(2,2,1), imshow(uint8(g1)), title('g1 236x300x3')
g2 = defocusing(w,r,'diffusion',@h_disk); %276x340x3
subplot(2,2,2), imshow(uint8(g2)), title('g2 276x340x3')
%Обратная задача - реконструкция (рефокусирование)
%дефокусированного изображения методом итераций Фридмана:
mg2 = size(g2,1); %=276 - кол-во строк
ng2 = size(g2,2); %=340 - кол-во столбцов
g2r = g2(1+r:mg2-r,1+r:ng2-r,:); %вырезка 256x320x3 из g2
w_K1 = refocusingF(g2r,g2r,r,@h_disk,0.3,5); %256x320x3
sig_w_K1 = rmsd(w_K1,w); %СКО_отн1=0.1674
```



Рис. 2.71. Реконструированные изображения методом итераций Фридмана при  $k = k_{\text{opt}}$ :

$$\begin{aligned} w_1 - w_0 &= 0, k_{\text{opt}} = 27, \\ w_2 - w_0 &= \nu \cdot g2r, k_{\text{opt}} = 27, \\ w_3 - w_0 &= g2r, k_{\text{opt}} = 28, \\ w_4 - w_0 &= 0.7\bar{w}, k_{\text{opt}} = 27 \end{aligned}$$

```
subplot(2,2,3), imshow(uint8(w_K1))
title('w_K1 256x320x3, w0=g2r, K=5, \sigma_{rel}=0.1674')
w_K2 = refocusingF(g2r,0.7*w,r,@h_disk,0.3,10); %256x320x3
sig_w_K2 = rmsd(w_K2,w); %CK0_отн2=0.0517
subplot(2,2,4), imshow(uint8(w_K2))
title('w_K2 256x320x3, w0=0.7w, K=10, \sigma_{rel}=0.0517')
```

На рис. 2.72 представлены результаты реконструкции методом итераций Фридмана сильно дефокусированного цветного изображения. Видим, что удается хорошо восстановить изображение лишь при  $w_0 = 0.7\bar{w}$ , т.е. при начальном приближении, весьма близком к точному изображению  $\bar{w}$ .

**О сравнительных возможностях редактора PhotoShop [108].** Нередко приходится слышать, что, мол, прямые и обратные задачи обработки смазанных, дефокусированных и зашумленных изображений успешно решаются в графическом редакторе Adobe PhotoShop, а также Paint Shop Pro, PhotoImpact и т.п. ([51,90,98,138] и др.), причем значительно проще, чем в системе MatLab, поскольку в редакторе PhotoShop не нужно программировать, а все операции можно выполнить с помощью "мыши". Однако анализ сравнительных возможностей редактора PhotoShop и пакета IPT си-

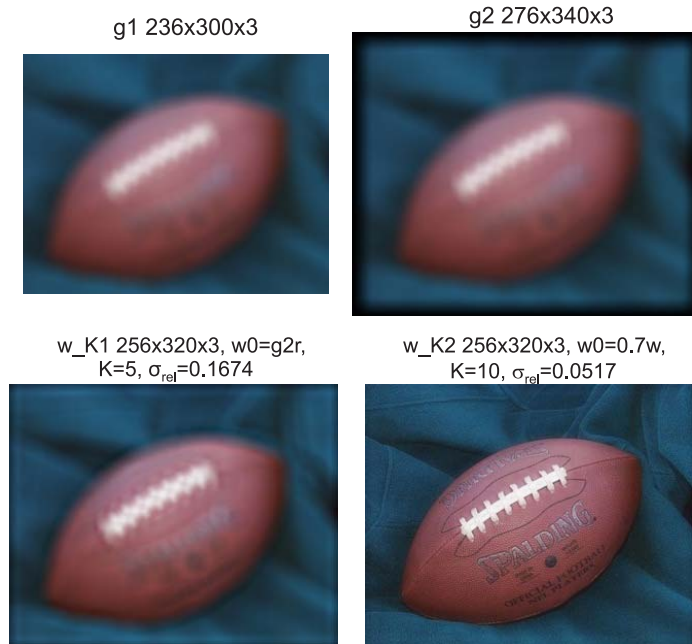


Рис. 2.72. Дефокусирование и рефокусирование методом итераций Фридмана цветного изображения `football.jpg`:  
`g1` — дефокусированное изображение с усечением согласно (2.70),  
`g2` — дефокусированное изображение с размытием краев согласно (2.71),  
`w_K1` — рефокусированное изображение при  $w_0 = g2r$ ,  
`w_K2` — рефокусированное изображение при  $w_0 = 0.7\bar{w}$

стемы MatLab, а также методики, изложенной в данной главе, показывает, что дела обстоят иначе.

Сначала рассмотрим, какие *прямые задачи* по смазыванию, дефокусированию и зашумлению изображений можно решить в рамках редактора PhotoShop (а также Paint Shop Pro, PhotoImpact и т.п.). В рамках меню (окна) "Фильтры" можно реализовать, например, следующие операции (фильтры):

- **Размытие в движении (Motion Blur)** — данный фильтр позволяет выполнить, в частности, размытие изображения под некоторым углом (Angle) на несколько пикселей (Distance). Это аналогично операции *смазывания* изображения в системе MatLab, например, `h=fspecial('motion'...); J=imfilter(I,h...);`

- **Размытие по Гауссу (Gaussian Blur)** — фильтр размывает изображение по гауссиане с параметром Radius. Это аналогично операции *дефокусирования* изображения в системе MatLab, например, `h=fspecial('gaussian'...); J=imfilter(I,h...);`

- **Добавить шум (Add Noise)** — добавляет к изображению либо равномерный, либо гауссовый шум с параметром Amount (определяет степень зашумленности изображения). Это аналогично *зашумлению* изображения в системе MatLab с помощью `m`-функции `imnoise.m`.

В общем, за исключением небольших отличий, PhotoShop, как и MatLab, позволяет решать прямую задачу — задачу искажения изображения (смазывание, дефокусирование, зашумление).

Однако гораздо более важной (и сложной) является *обратная задача* — задача восстановления изображения по искаженному изображению. И в этом отношении возможности редактора PhotoShop сильно ограничены по сравнению с системой MatLab. В редакторе PhotoShop можно отметить следующие фильтры, позволяющие в той или иной мере решить обратную задачу:

- **Резкость (Sharpen)** — данный фильтр призван фокусировать расплывчатые изображения, повышая контраст смежных пикселей. Однако его использование, как показывает обработка ряда дефокусированных изображений, практически не улучшает фокусировку изображений, как это можно сделать в рамках системы MatLab (см. рис. 2.59 и 2.61) с помощью m-функций `deconvwnr`, `refocusingT` и др.

- **Медиана (Median)** — данный фильтр призван уменьшать шум в изображении (рекомендуется также для устранения или уменьшения эффекта движения на изображении). Проверка показывает, что фильтр действительно уменьшает шум в изображении, но при этом искажает и само изображение. Использование же таких m-функций системы MatLab, как `wiener2` (фильтрация шума методом адаптивной винеровской фильтрации), `medfilt2` (фильтрация шума медианным фильтром) и `ordfilt2` (фильтрация шума ранговым фильтром), является более эффективным (см. рис. 2.25). Что же касается применения фильтра Медиана (Median) к смазанным изображениям с целью устранения или уменьшения смазывания, то это совершенно не дает нужного эффекта в противоположность тому, что можно получить с помощью m-функций системы MatLab `deconvwnr` (устранение смазывания методом параметрической фильтрации Винера), `desmearingf` и `desmearingq` (устранение смазывания методом ПФ или квадратур с регуляризацией Тихонова) (см. рис. 2.21, 2.35, 2.38, 2.43, 2.44, 2.47).

Итак, использование редактора PhotoShop (а также Paint Shop Pro, PhotoImpact и т.п.) позволяет моделировать такие искажения изображений, как смазывание, дефокусирование и зашумление — *прямая задача*. При этом результаты моделирования искажений получаются примерно такими же, как и при использовании m-функций системы MatLab. Что же касается *обратной задачи* — восстановления изображений по искаженным изображениям, то возможности редактора PhotoShop сильно ограничены по сравнению с системой MatLab. Это можно объяснить тем, что в системе MatLab для решения обратной задачи привлечен мощный математический аппарат (решение некорректных 1-мерных и 2-мерных интегральных уравнений Фредгольма I рода методами параметрической фильтрации Винера, регуляризации Тихонова, алгоритмом Люси–Ричардсона и др.), а в редакторе PhotoShop использованы лишь такие довольно простые математические методы, как медианная фильтрация.

### Контрольные задания и вопросы

1. Изложите постановку задачи реконструкции дефокусированных изображений.
2. Дайте вывод основного (интегрального) соотношения задачи дефокусирования

изображения. В чем состоит схожесть и различие соотношений (2.54) и (2.55)?

3. В чем заключается нестандартность интегрального уравнения (2.55)? Преобразуйте его к стандартной форме.

4. Что такое функция рассеяния точки (ФРТ) в задаче дефокусирования изображения? Чем отличаются ФРТ (2.58) и (2.59)?

5. Что означает, что ФРТ является пространственно-инвариантной? Приведите примеры устройств наблюдения, ФРТ которых является пространственно-инвариантной.

6. Изложите прямую задачу дефокусирования изображения. Каким интегральным выражением она описывается? Чем вызвана необходимость введения так называемых "граничных условий"? Опишите их типы.

7. Сформулируйте суть приема усечения дефокусированного изображения. В чем его отличие от приема "граничных условий"?

8. В чем заключается Соглашение о координатах, принятое в данной книге?

9. Дайте сравнительную характеристику выражениям (2.65), (2.67), (2.68) и (2.70).

10. Для чего вводится прием размытия краев дефокусированного изображения? Дайте определение эффекта Гиббса и объясните его причину. Охарактеризуйте схему на рис. 2.58. В чем состоит отличие выражения (2.71) от (2.70)?

11. Разъясните  $m$ -функцию `defocusing.m`.

12. Изложите обратную задачу реконструкции дефокусированных изображений. Каким интегральным уравнением она описывается?

13. Изложите кратко метод инверсной фильтрации решения уравнения (2.56). Почему этот метод неустойчив? В чем состоит идея псевдоинверсной фильтрации?

14. Дайте краткую характеристику методов оптимальной и параметрической фильтрации Винера, а также регуляризации Тихонова на основе соотношений (2.56), (2.77)–(2.85).

15. Разъясните внешнюю  $m$ -функцию `deconvnr` и собственную  $m$ -функцию `refocusingT`.

16. Объясните различие результатов реконструкции, приведенных на рис. 2.62, 2.64 и 2.67.

17. В чем состоит достоинство и недостаток решения уравнения (2.56) методом 2-мерного ПФ (с использованием методов Винера, Тихонова и т.д.)? В чем состоят трудности решения уравнения (2.56) методом квадратур (кубатур)?

18. Дайте краткую характеристику метода итеративной регуляризации Фридмана на основе соотношений (2.87)–(2.93).

19. Разъясните  $m$ -функцию `refocusingF`. Охарактеризуйте результаты, приведенные на рис. 2.69–2.72. В чем состоят трудности реализации метода итераций Фридмана?

20. Кратко сформулируйте возможности графического редактора PhotoShop в решении прямой и обратной задач обработки смазанных, дефокусированных и зашумленных изображений. Сравните с аналогичными возможностями системы MatLab.

## Глава 3

# ОБРАТНЫЕ ЗАДАЧИ СПЕКТРОСКОПИИ

В этой главе рассмотрим задачу восстановления непрерывных и дискретных спектров путем их математической и компьютерной обработки. Это будет еще одно приложение аппарата интегральных уравнений к решению прикладной задачи — обратной задачи спектроскопии.

### 3.1. Введение в спектральный анализ

**Спектральный анализ и приборы.** Для качественного и количественного исследования веществ широко используется *спектральный анализ* [59, 102, 114]. Он заключается [128, с. 708] в определении состава вещества (а также его температуры, плотности, давления и т.д.) на основе изучения его спектров — спектров излучения (эмиссии), поглощения (абсорбции), отражения, комбинационного рассеяния света и люминесценции [128, с. 303, 354, 713], [177]. При этом под *спектром* подразумевается зависимость интенсивности излучения от частоты  $\nu$  или от длины волны  $\lambda$ .<sup>17</sup>

Для разложения излучения в спектр и его регистрации используются *оптические спектральные приборы* [67, с. 309–312], [114, с. 23], [128, с. 703–708]. Каждый такой прибор состоит из трех основных частей: осветительной (необязательная часть), спектральной и регистрирующей. В зависимости от способа регистрации спектра различают следующие *приборы*:

- *спектроскопы* (с визуальной регистрацией);
- *спектрографы* (с фоторегистрацией);
- *спектрометры* (например, интерферометр Фабри–Перо [67, с. 127–129]);
- *спектрофотометры* (с фотоэлектрической регистрацией) [128, с. 710–711].

**Типы спектров.** Спектры бывают (см. рис. 3.1) [67, с. 647–648], [128, с. 712–713]:

- *непрерывные* (охватывающие широкий диапазон частот), или *сплошные* (примеры: спектры расплавленного металла, Солнца, горячего газового потока в сопле ракеты, в выхлопной трубе автомобиля и т.д.);

<sup>17</sup>Часто вместо частоты  $\nu = c/\lambda$  используют  $k = 1/\lambda$  — *волновое число* [128, с. 85], [153].

- *дискретные*, или *линейчатые*, состоящие из отдельных (мономатических) спектральных линий, соответствующих дискретным значениям частоты  $\nu$  (примеры: спектры атомарного водорода, натрия, меди, спектры веществ, находящихся в глубоком вакууме, в частности, спектры рассеянных межзвездных туманностей);
- *полосатые*, состоящие из ряда полос, каждая из которых, в свою очередь, состоит из набора близких дискретных линий (пример: спектр излучения паров иода, а также спектры веществ, находящихся в неглубоком вакууме, в частности, спектры люминесцентных ламп и ламп "дневного света"), и
- *сложные*, или *комбинированные*, состоящие, например, из непрерывного спектра и ряда дискретных линий.

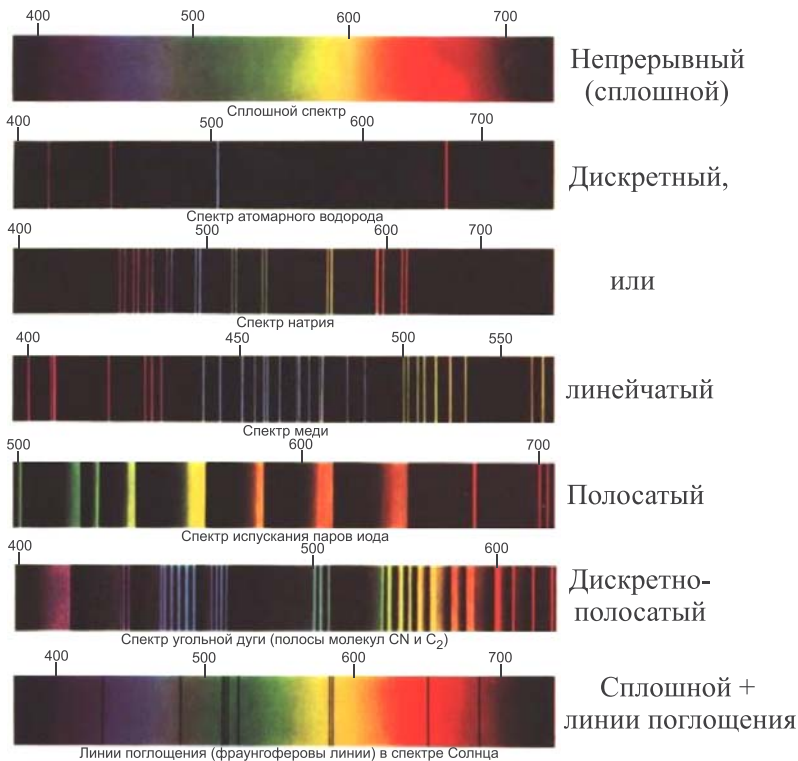


Рис. 3.1. Типы спектров (указаны длины волн  $\lambda$  в нм)  
[128, с. 528–529]

См. также далее рис. 3.2, 3.4, 3.6, 3.7, 3.9, 3.10 — непрерывные спектры и 3.11–3.14 — дискретные спектры.

Строго говоря, отдельная спектральная линия не является дискретной линией (мономатической) с вполне определенным значением  $\lambda$ , поскольку имеет, во-первых, минимальную (естественную) ширину, обусловленную квантовыми эффектами, во-вторых, ширину, обусловленную эффектами Доплера (тепловое уширение), Зеемана (магнитное уширение), Штарка (электрическое уширение), дифракцией на диафрагмах оптической системы спектрального прибора и т. д. [67, с. 521], [114, с. 24], [128,



с. 713], [155]. Однако если линия имеет лишь естественную ширину, то обычно ее считают дискретной (монохроматической). Пример: линии спектров рассеянных межзвездных туманностей имеют практически лишь естественную ширину, так как тепловые, магнитные и электрические эффекты в космосе обычно чрезвычайно малы. Поэтому эти спектры, как правило, считают дискретными.

**Типы спектрального анализа.** Спектральный анализ можно разделить на широкополосный и узкополосный (а также атомный, молекулярный, эмиссионный, абсорбционный, качественный, количественный и т. д.) [102, с. 81], [128, с. 708].

*Широкополосная спектрометрия* — это изучение спектра в широкой области частот, например, изучение спектра звезды во всем видимом диапазоне (от красного до фиолетового) — см. рис. 3.1 и 3.2а.

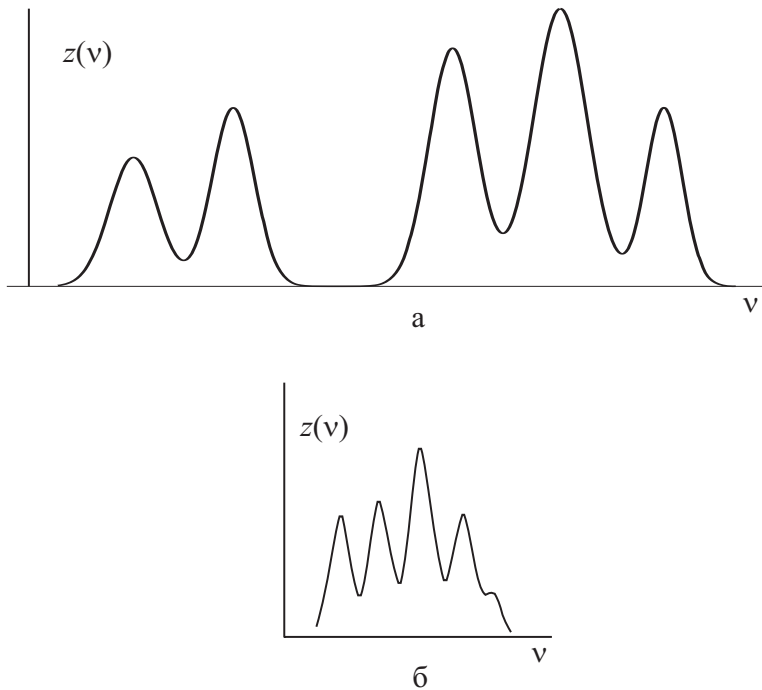


Рис. 3.2. Широкополосный (а) и узкополосный (б) спектр  $z(\nu)$

*Узкополосная спектрометрия* — это изучение спектра в узкой полосе частот, например, изучение сверхтонкой структуры мессбауэровской линии, обусловленной магнитными или электрическими полями и тепловыми эффектами [128, с. 407] — см. рис. 3.2б.

Однако деление на широко- и узкополосную спектрометрию зачастую условно.

**Области применения спектрального анализа.** *Областями применения* спектрального анализа являются:

- *физика* (изучение спектров газов, жидкостей, металлов и плазмы, исследование атомной сенсibilизированной флуоресценции смесей и паров металлов по контурам

спектральных линий [61]);

- *астрофизика* (изучение спектров планет, звезд, галактик, туманностей, комет, квазаров, реликтового излучения);
- *металлургия* (определение по спектру состояния расплавленного металла);
- *химия и ЯМР-спектроскопия* (определение по спектру химического состава вещества);
- *геофизика* (разведка руд, минералов, нефти, газа) и т. д.

Следует отметить и такое применение спектрального анализа, как определение магнитного или электрического поля по сверхтонкой структуре линии на основе эффекта Зеемана или Штарка.

Однако эффективность применения спектрального анализа зависит, в первую очередь, от того, насколько высокой является разрешающая способность спектрального прибора, а именно, разрешает ли прибор близкие линии, выделяет ли он слабые линии, определяет ли он сверхтонкую структуру линии. Если же разрешающая способность спектрального прибора является недостаточной, то и применение спектрального анализа будет неэффективным. Тем не менее, если спектральные измерения дополнить математико-компьютерной обработкой, то можно повысить разрешающую способность спектрального прибора, что мы и хотим показать в данной главе.

Количественно это можно выразить следующим образом. Часто для характеристики качества спектральных измерений используют так называемую *разрешающую способность спектрального прибора*, равную [94], [128, с. 704]

$$R = \lambda/\delta\lambda,$$

где  $\lambda$  – средняя длина волны рабочего диапазона, а  $\delta\lambda$  – спектральное разрешение прибора (примерно равно ширине его аппаратной функции). На современных спектрометрах удается достичь очень высокого значения  $R$ . В частности, спектрометры, использующие интерферометры Фабри–Перо, позволяют получать в видимой области  $R \sim 10^5 - 10^6$ . Например, в ИК-области  $R \approx 2 \cdot 10^5$  при  $\lambda \approx 3 \cdot 10^3$  нм [128, с. 705].

Однако это сложные, крупногабаритные и дорогие спектрометры, содержащие длиннофокусные монохроматоры, помещаемые в вакуумные корпуса в виброзащищенных и термостабилизированных помещениях. В них используется 2- или 4-кратная дифракция, применяются высокочувствительные охлаждаемые приемники и т. д. [128, с. 705].

Еще одним недостатком чисто инструментального (без математической обработки) получения спектра с высоким разрешением является то, что высокое разрешение связано с узостью щели прибора и, значит, со слабостью светового потока. Если же увеличивать ширину щели и тем самым увеличивать световой поток, то ухудшится разрешение [114, с. 31].

Поэтому актуальным является путь, когда измерение спектра выполняется на недостаточно совершенном спектрометре (у которого, например,  $R \sim 10^3 - 10^4$ ), но ширина щели и световой поток повышены, а затем выполняется математико-компьютерная обработка спектра (некоторым эффективным методом) с целью повышения  $R$  (до  $\sim 10^5 - 10^6$ ).

Говоря далее о спектральном анализе, мы будем иметь в виду, в основном, спектры излучения, а также их измерения с помощью спектрометров и их обработку, хотя нижеизложенное справедливо и для иных типов спектров и приборов.

### Контрольные задания и вопросы

1. Дайте определение спектрального анализа и спектра.
2. Перечислите оптические спектральные приборы.
3. Назовите типы спектров. От чего зависит тип спектра?
4. Можно ли строго считать отдельную спектральную линию дискретной, т. е. имеющей определенную частоту?
5. Что такое широкополосный и узкополосный спектральный анализ? Приведите примеры.
6. Назовите области применения спектрального анализа.
7. От чего зависит, в первую очередь, эффективность применения спектрального анализа?

## 3.2. Математическое описание задачи спектроскопии в случае непрерывного спектра

Перейдем к математическому описанию прямой и обратной задач спектроскопии в случае, когда спектр является непрерывным (сплошным) [33, 59, 94, 102, 114, 128]. Введем такие понятия, как аппаратная функция (АФ) спектрометра, истинный и измеренный спектр и др.

**Основные понятия и определения.** Введем следующее

**Определение.** *Аппаратной функцией* — АФ (или спектральной чувствительностью — СЧ, инструментальным контуром, функцией пропускания, функцией цели, частотной характеристикой — ЧХ) спектрометра [33, 94], [102, с. 82], [114, с. 25], [128, с. 704], [153]  $K(\nu, \nu')$  называется реакция спектрометра (в виде измеренной интенсивности) на дискретную линию единичной интенсивности и частоты  $\nu'$  при настройке спектрометра на частоту  $\nu$  — см. рис. 3.3.

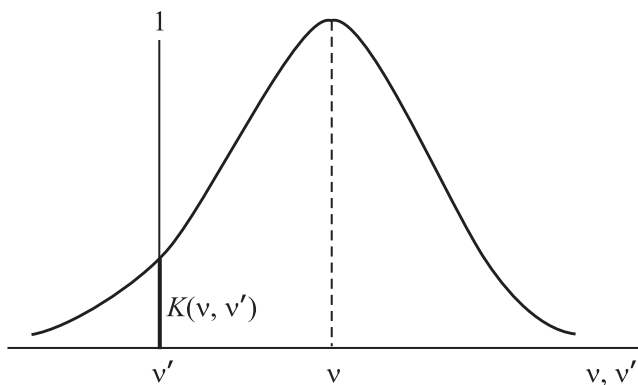


Рис. 3.3. К определению АФ спектрометра  $K(\nu, \nu')$

Фиксируя  $\nu'$  и изменяя  $\nu$ , получим некоторую кривую  $K(\nu, \nu')$  (см. рис. 3.3). Аналогичные кривые получим для других значений  $\nu'$ . В результате получим 2-мерную функцию  $K(\nu, \nu')$ .

Пример АФ, представленной аналитической формулой в виде гауссианы:

$$K(\nu, \nu') = \frac{1}{\sqrt{2\pi}a} \exp\left(-\frac{(\nu - \nu')^2}{2(a - b\nu)^2}\right) \quad (3.1)$$

с максимумом при  $\nu' = \nu$  и с эффективной шириной (с дисперсией  $\sigma^2 = (a - b\nu)^2$ ), обратно пропорциональной  $\nu$  ( $a > 0$  и  $b > 0$  — некоторые константы), т. е. с повышением частоты АФ становится уже — эффект, характерный для широкополосной спектроскопии (другие примеры АФ см. в [94], [114, с. 28–32], [153, 155]).

Если же форма  $K(\nu, \nu')$  практически не меняется с изменением  $\nu$  (в пределах полосы измерений), то  $K(\nu, \nu') = K(\nu - \nu')$ , т. е. АФ является разностной функцией — эффект, характерный для узкополосной спектроскопии.

АФ спектрометра полагается известной. Она должна быть представлена в паспорте спектрометра или определена, например, с помощью одночастотного лазера, стабилизированного по провалу Лэмба [61].

Далее обозначим через  $z(\nu)$  *истинный спектр* — зависимость интенсивности излучения от частоты  $\nu$ , неискаженная никакими мешающими факторами (конечной шириной АФ спектрометра, шумами и т. д.), а через  $u(\nu)$  — *измеренный* (экспериментальный) *спектр* — это спектр, измеренный путем сканирования по частоте (в некотором диапазоне частот) реальным спектрометром, а именно, с учетом его АФ  $K(\nu, \nu')$  и при наличии различных шумов (погрешностей измерений, турбулентностей в среде, через которую идет излучение, и т. д.). С этих позиций, истинный спектр  $z(\nu)$  — это спектр, который был бы измерен при бесконечно узкой (идеальной) АФ и в отсутствие шумов.

Измеренный спектрометром спектр  $u(\nu)$  обычно отличается от истинного спектра  $z(\nu)$ . Это проявляется, во-первых, в большей сглаженности  $u(\nu)$  по сравнению с  $z(\nu)$  (неразрешены близкие линии, заглажена тонкая структура некоторых линий) и, во-вторых, в зашумленности  $u(\nu)$  (слабые линии "тонут" в шуме). На рис. 3.4 даны: истинный спектр  $z(\nu)$  (приведенный на рис. 3.2а) и соответствующий ему измеренный (экспериментальный) спектр  $u(\nu)$ , искаженный спектральной чувствительностью (на рис. 3.4 приведено одно из "сечений" 2-мерной функции  $K(\nu, \nu')$  при некотором фиксированном  $\nu$ ). Приведен также измеренный зашумленный спектр

$$\tilde{u}(\nu) = u(\nu) + \delta u + F, \quad (3.2)$$

где  $\delta u$  — случайная шумовая компонента (обусловленная погрешностями измерений и внешними помехами), а  $F$  — детерминированная шумовая компонента (фон), полагаемая постоянной.

Из рис. 3.4 видно, что вследствие конечной ширины АФ близкие линии (1-я и 2-я, а также 3-я, 4-я и 5-я, считая слева направо) не разрешены в  $u(\nu)$ , тем более, в  $\tilde{u}(\nu)$ . Данный эффект тем сильнее, чем шире АФ спектрометра  $K(\nu, \nu')$ , другими словами, чем меньше разрешающая способность спектрометра.

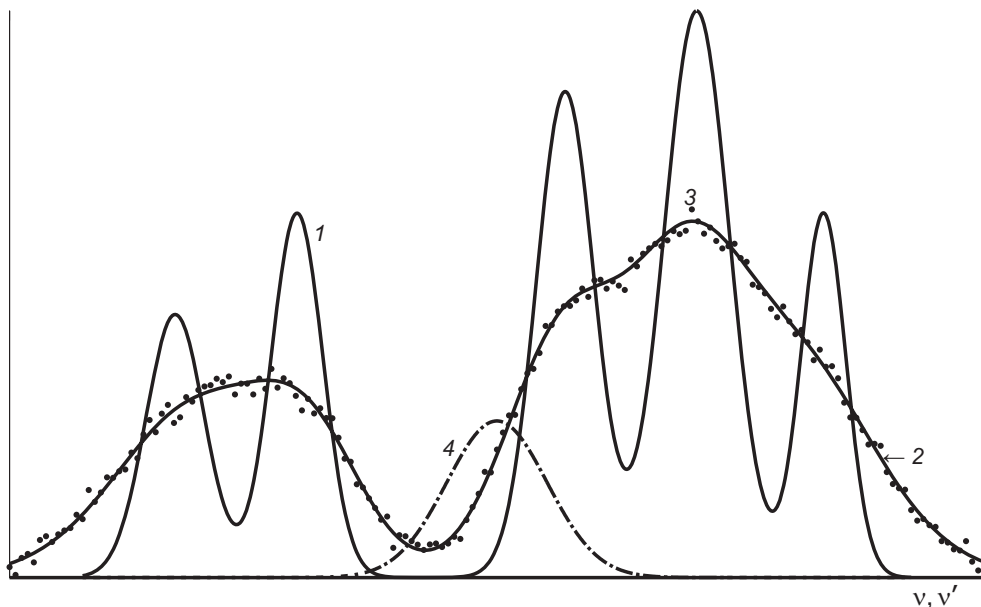


Рис. 3.4. 1 — истинный спектр  $z(\nu)$  (см. рис. 3.2),  
 2 — измеренный без шума спектр  $u(\nu)$ ,  
 3 — измеренный зашумленный спектр  $\tilde{u}(\nu)$ ,  
 4 — АФ  $K(\nu, \nu')$  при некотором фиксированном  $\nu$

В принципе такая же ситуация имеет место в случае непрерывного узкополосного спектра (см. рис. 3.2б), например, в случае сверхтонкой структуры отдельной линии.

**Задача редукции к идеальному спектральному прибору.** Возникает следующая задача: по экспериментальному спектру  $\tilde{u}(\nu)$  и АФ  $K(\nu, \nu')$  путем математической и компьютерной обработки восстановить истинный спектр  $z(\nu)$ . Успешное решение этой задачи позволит повысить разрешающую способность спектрометра, а значит, качество спектрального анализа (например, более точно определить фазовое состояние расплавленного металла в дожде или химический состав космического объекта — туманности, кометы, поверхности звезды и т. д.). Задача определения истинного спектра  $z(\nu)$  по экспериментальному спектру  $\tilde{u}(\nu)$  и АФ  $K(\nu, \nu')$  называется *задачей редукции* (приведения) *профиля спектра к идеальному спектральному прибору* (другое название: *исключение АФ*) [17, 61, 94].

Чтобы повысить разрешающую способность спектрометра, можно использовать (ср. [114, с. 27]) физико-техничко-коммерческий путь, а именно, использовать более совершенный (а, значит, более дорогой) спектрометр. Но можно использовать и математико-компьютерный путь, а именно, выполнить математическую обработку (с помощью компьютера) результатов измерений путем решения соответствующего уравнения. Это — один из вариантов известной редукционной проблемы Рэля [33].

**Математическая формулировка задачи спектроскопии.** Рассмотрим случай *непрерывного спектра*, когда искомым спектром  $z(\nu)$ ,  $\nu \in [a, b]$ , есть кусочно-непрерывная функция (см. рис. 3.2, 3.4). Этот случай имеет место обычно для веществ с

повышенной плотностью (расплавленный металл и т.д.) или при изучении сверхтонкой структуры линии, когда пределы  $[a, b]$  узкие. В этом случае измеренное значение интенсивности  $u(\nu)$  при настройке спектрометра на частоту  $\nu$  равно интегралу (сумме) по всем интенсивностям  $z(\nu')$  с весовой функцией, равной  $K$ , т.е. (ср. [114, с. 25])

$$u(\nu) = \int_a^b z(\nu') K(\nu, \nu') d\nu'. \quad (3.3)$$

Из (3.3), варьируя  $\nu$  (т.е. выполняя сканирование по спектру) и учитывая зашумленность  $u(\nu)$  (см. (3.2)), получим (поменяв местами левую и правую части в (3.3)):

$$\int_a^b K(\nu, \nu') z(\nu') d\nu' = \tilde{u}(\nu), \quad c \leq \nu \leq d, \quad (3.4)$$

где  $[c, d]$  — пределы изменения  $\nu$  (обычно более широкие, чем  $[a, b]$ , см. рис. 3.4).

В соотношении (3.4) известны (измерены или заданы)  $\tilde{u}(\nu)$ ,  $K(\nu, \nu')$ ,  $a$ ,  $b$ ,  $c$ ,  $d$ , а  $z(\nu')$  является искомой функцией. Соотношение (3.4) есть интегральное уравнение Фредгольма I рода относительно  $z(\nu')$ . Математические названия входящих в (3.4) функций:  $K(\nu, \nu')$  — ядро (физическое название: АФ),  $z(\nu')$  — искомая функция (истинный спектр),  $\tilde{u}(\nu)$  — правая часть (измеренный спектр).

Если  $K(\nu, \nu') = K(\nu - \nu')$ , то (3.4) обычно записывается в виде:

$$\int_0^{\infty} K(\nu - \nu') z(\nu') d\nu' = \tilde{u}(\nu), \quad 0 \leq \nu < \infty. \quad (3.5)$$

Соотношение (3.5) есть ИУ Фредгольма I рода типа свертки на полуоси. Задача решения уравнений (3.4) и (3.5) является некорректной (в первую очередь, сильно неустойчивой). На рис. 3.5 приведен результат численного решения ИУ Фредгольма I рода типа свертки (3.5) методом преобразования Фурье без дополнительного использования какого-либо устойчивого метода. Видим, что решение получилось чрезвычайно неустойчивым — в виде так называемой "пилы", ничего общего не имеющей с точным решением  $\bar{z}(\nu)$ .<sup>18</sup>

Поэтому для устойчивого решения ИУ (3.4) и (3.5) необходимо применение устойчивых методов, например, метода регуляризации Тихонова [22, 37, 38, 59, 102, 117, 121, 151] аналогично решению уже рассмотренных выше 1-мерных ИУ (2.13) и (2.30).

Если пределы  $[a, b]$  широкие, то мы имеем дело с обработкой спектра в широкой полосе частот и в этом случае нужно учитывать изменение формы  $K(\nu, \nu')$  с изменением  $\nu$  (обычно  $K(\nu, \nu')$  становится более узкой с ростом  $\nu$ ). Если же пределы  $[a, b]$

<sup>18</sup>Мы специально не приводим численных параметров метода (шаг дискретизации и др.), а также выражение для  $K(\nu)$ , так как такое пилообразное решение получается практически всегда. Таким же неустойчивым получается решение и другим неустойчивым методом, например, методом квадратур [102, с. 182], [117, с. 162].

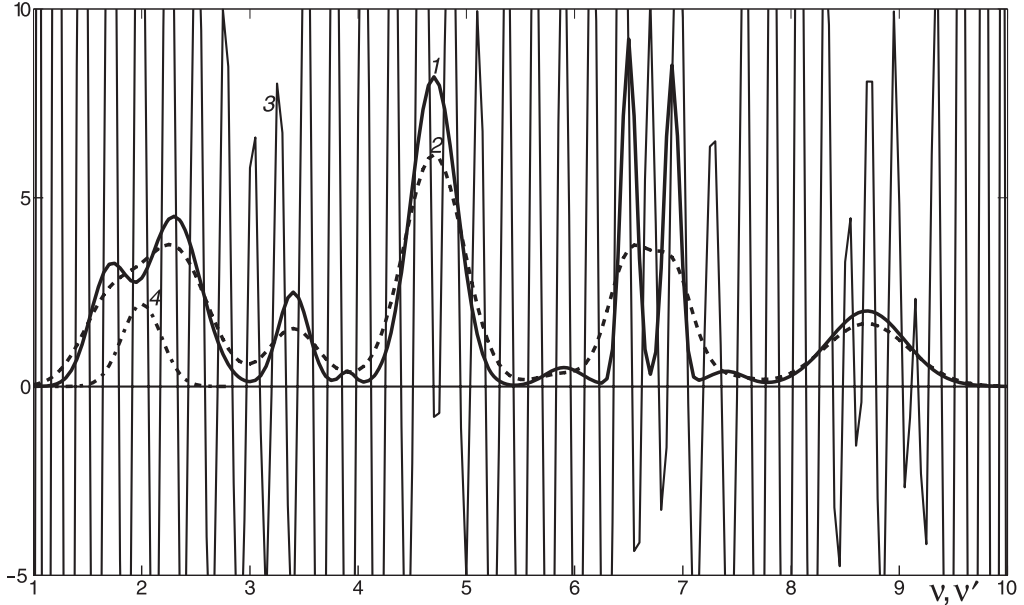


Рис. 3.5. 1 — истинный спектр  $\bar{z}(\nu)$ , 2 — сглаженный спектр  $u(\nu)$ ,  
3 — неустойчивое решение ("пила"), 4 — АФ  $K(\nu)$

узкие (например, при изучении тонкой структуры одной линии), то это будет задача обработки спектра в узкой полосе частот и следует использовать ИУ вида:

$$\int_a^b K(\nu - \nu') z(\nu') d\nu' = \tilde{u}(\nu), \quad c \leq \nu \leq d, \quad (3.6)$$

— ИУ Фредгольма I рода с разностным ядром.

**Метод квадратур с регуляризацией Тихонова.** Приведем следующий упрощенный (но достаточно эффективный) вариант решения ИУ Фредгольма I рода методом квадратур с регуляризацией Тихонова. Запишем ИУ Фредгольма I рода в общем виде (ср. [22, с. 240], [102, с. 194]):

$$\int_a^b K(x, s) y(s) ds = f(x), \quad c \leq x \leq d, \quad (3.7)$$

где  $K(x, s)$  — ядро,  $y(s)$  — искомая функция,  $f(x)$  — правая часть, все функции вещественные.

Пусть правая часть  $f(x)$  задана на следующей, вообще говоря, неравномерной  $x$ -сетке узлов:

$$c = x_1 < x_2 < \dots < x_i < \dots < x_m = d, \quad (3.8)$$

а решение  $y(s)$  ищется на другой, вообще говоря, неравномерной  $s$ -сетке узлов:

$$a = s_1 < s_2 < \dots < s_j < \dots < s_n = b, \quad (3.9)$$

где  $i$  и  $j$  — номера узлов, причем  $m \leq n$ .

Распишем интеграл в (3.7) по формуле трапеций, получим систему  $m$  линейных алгебраических уравнений (СЛАУ) относительно  $n$  неизвестных  $y_j \equiv y(s_j)$ :

$$\sum_{j=1}^n r_j K_{ij} y_j = f_i, \quad i = \overline{1, m}, \quad (3.10)$$

где

$$r_1 = \frac{s_2 - s_1}{2}, \quad r_j = \frac{s_{j+1} - s_{j-1}}{2}, \quad j = \overline{2, n-1}, \quad r_n = \frac{s_n - s_{n-1}}{2} \quad (3.11)$$

— коэффициенты квадратурной формулы трапеций,  $K_{ij} \equiv K(x_i, y_j)$ ,  $f_i \equiv f(x_i)$ . Запишем СЛАУ (3.10) в виде (типа (2.37)):

$$A y = f, \quad (3.12)$$

где  $a_{ij} = r_j K_{ij}$ ,  $i = \overline{1, m}$ ,  $j = \overline{1, n}$ , — элементы матрицы  $A_{m \times n}$ ,  $y_{n \times 1}$  — искомый вектор-столбец,  $f_{m \times 1}$  — правая часть (вектор-столбец).

В методе регуляризации 0-го порядка Тихонова вместо СЛАУ (3.12) решается СЛАУ [22, с. 238, формула (4.51)], [102, с. 193, формула (8.6)]:

$$(\alpha E + A^T A) y_\alpha = A^T f, \quad (3.13)$$

где  $\alpha > 0$  — параметр регуляризации,  $E$  — единичная матрица (диагональные элементы — единицы, остальные — нули),  $A^T$  — транспонированная матрица. Регуляризованное решение  $y_\alpha$  можно записать в виде (ср. (2.38)):

$$y_\alpha = (\alpha E + A^T A)^{-1} A^T f. \quad (3.14)$$

В рамках системы MatLab 7 разработана собственная m-функция Tikh.m:

```
%Tikh.m, m-функция, MatLab7
%Решение интегрального уравнения Фредгольма I рода
%методом квадратур с регуляризацией Тихонова.
%Входные параметры (аргументы):
%f - правая часть f(x) - вектор-строка, 1xm double,
%x - сетка узлов по x (необязательно равномерная), 1xm double,
%s - сетка узлов по s (необязательно равномерная), 1xn double,
%kern - ядро K(x,s) интегрального уравнения (m-функция), double,
%alpha>0 - параметр регуляризации, double.
%Выходные параметры (аргументы):
%y_alpha - регуляризованное решение y(alpha), double,
%y0_alpha - регуляризованное решение y(alpha)>=0, double.
%Возможные варианты обращений к Tikh:
%y_alpha = Tikh(f,x,s,@kern,alpha);
%или
%[y_alpha,y0_alpha] = Tikh(f,x,s,@kern,alpha);
```



```

function [y_alpha,y0_alpha] = Tikh(f,x,s,kern,alpha)
m = length(x); %длина вектора x 1xm
n = length(s); %длина вектора s 1xn
%Формируем вектор-строку квадратурных коэффициентов 1xn:
r = [s(2)-s(1), s(3:n)-s(1:n-2), s(n)-s(n-1)]/2;
%Формируем матрицу A mxn:
A = repmat(r,m,1).*kern(repmat(x',1,n),repmat(s,m,1));
%Решение ИУ в виде решения СЛАУ:
y_alpha = ((alpha*eye(n)+A'*A)\(A'*f'))'; %y_alpha 1xn
if (nargout==2)
    y0_alpha = y_alpha;
    y0_alpha(y0_alpha<0) = 0; %доп. условие: y>=0
end %if

```

При этом важным является вопрос о *выборе параметра регуляризации*  $\alpha$ . Существует ряд способов его выбора [22, 28, 29, 76, 102, 103, 117, 121, 151, 158]: способ невязки, обобщенный принцип невязки, способ отношения, метод перекрестной значимости, локальный регуляризирующий алгоритм, способ подбора и др.

В гл. 2 при реконструкции изображений были использованы еще два способа выбора  $\alpha$ : по минимуму погрешности реконструкции  $\sigma_{\text{rel}}(\alpha)$  (2.41) (когда решается модельный пример с известным точным изображением) и по визуальной оценке регуляризованного (реконструированного) изображения  $w_\alpha$ .

В задаче спектроскопии использовать второй способ (по визуальной оценке восстановленного спектра  $z_\alpha$ ) не представляется возможным, так как практически нет четких критериев оценки качества восстановления спектра. Что же касается первого способа (по минимуму погрешности реконструкции спектра), то предлагаем реализовать его в рамках *способа модельных (эталонных) примеров* [110] (другие названия: способ моделирования [22, с. 246–249], [99, 100], метод вычислительных экспериментов [23, 77], способ псевдообратного оператора [110]). Данный способ использует дополнительную (априорную) информацию о решении (искомом спектре).

В работах [22, с. 239–240], [23, 99, 100, 110] и др. дано математическое обоснование способа модельных примеров. Приведем основные положения этого обоснования.

Рассмотрим операторное уравнение I рода:

$$Ay = f, \quad y \in Y, \quad f \in F,$$

где  $y$  – искомый, а  $f$  – заданный элементы гильбертовых пространств  $Y$  и  $F$ ;  $A$  – линейный ограниченный оператор из  $Y$  в  $F$ . Оператор  $A$  не предполагается непрерывно обратимым, поэтому задача решения этого уравнения является *некорректной*. Полагаем, что вместо точных  $f$  и  $A$  известны их приближения  $\tilde{f}$  и  $\tilde{A}$  такие, что  $\|\tilde{f} - f\| \leq \delta$ ,  $\|\tilde{A} - A\| \leq \xi$ . Частным случаем данного операторного уравнения является ИУ Фред-

гольма I рода (см. (3.7)):

$$Ay \equiv \int_a^b K(x, s) y(s) ds = f(x), \quad c \leq x \leq d.$$

В методе регуляризации 0-го порядка Тихонова решается уравнение Эйлера (ср. (3.13)) [22, с. 238], [121, с. 11]:

$$\alpha y_\alpha + \tilde{A}^* \tilde{A} y_\alpha = \tilde{A}^* f.$$

Пусть  $\Delta y_\alpha = y_\alpha - y$  — погрешность регуляризованного решения  $y_\alpha$ , а  $y$  — точное решение (в качестве такового берем нормальное псевдорешение [22, 76, 117, 121]). Обозначим  $\tilde{R} = \tilde{A}^* \tilde{A}$ . Получена [22, с. 239], [99, 100] следующая оценка относительной погрешности регуляризованного решения по норме:

$$\frac{\|\Delta y_\alpha\|}{\|y\|} \leq \varepsilon(\alpha),$$

где

$$\varepsilon(\alpha) = \frac{\|\tilde{A}\|}{2\sqrt{\alpha}} \eta + \frac{p\alpha}{p\alpha + 1},$$

причем  $\eta = \delta_{\text{отн}} + \xi_{\text{отн}}$ , где  $\delta_{\text{отн}} = \delta/\|f\|$ ,  $\xi_{\text{отн}} = \xi/\|A\|$  — относительные погрешности исходных данных,  $p = \|\tilde{A}^+\|^2 = \|\tilde{R}^+\|$ ,  $A^+$  — псевдообратный оператор:  $A^+ f = y$  [76, с. 184]. Функция  $\varepsilon(\alpha)$  является верхней огибающей для истинной относительной погрешности (ср. далее (3.18))

$$\sigma_{\text{rel}}(\alpha) = \frac{\|\Delta y_\alpha\|}{\|y\|}.$$

Показано [22, с. 239–240], [99, 100], что функция  $\varepsilon(\alpha)$  имеет (единственный) минимум при  $p \cdot (\|\tilde{A}\| \eta)^2 < 27/16 \approx 1.69$ .

Приведенные соотношения показывают, что оценка относительной погрешности  $\|\Delta y_\alpha\|/\|y\|$  регуляризованного решения  $y_\alpha$  зависит от  $\tilde{A}$  и  $\eta$ . Поэтому, если решается несколько задач (примеров) с одинаковыми  $\tilde{A}$  и  $\eta$ , то для них ход функции  $\varepsilon(\alpha)$ , дающей верхнюю оценку погрешности  $\sigma_{\text{rel}}(\alpha)$ , будет одинаковый.

Отсюда следует, что при решении некоторого *исходного примера*  $P$  с неизвестным (искомым) решением  $y_P$  можно использовать результаты решения другого, *модельного, примера*  $Q$  с известным (заданным) точным решением  $y_Q$ , но с такими же  $\tilde{A}$  и  $\eta$ , что и у примера  $P$ . При решении примера  $Q$  можно рассчитать функцию

$$\sigma_{\text{rel}}(\alpha)_Q = \frac{\|\Delta y_{\alpha Q}\|}{\|y_Q\|}$$

и по ней найти  $\alpha_{\text{opt } Q}$  — значение  $\alpha$ , при котором  $\sigma_{\text{rel}}(\alpha) = \min_{\alpha} \sigma_{\text{rel}}(\alpha)_Q$ . Это значение  $\alpha_{\text{opt } Q}$  может быть использовано при решении исходного примера  $P$ .

Отметим также, что получена [22, с. 249] следующая оценка абсолютной погрешности регуляризованного решения:

$$\|\Delta y_\alpha\| \leq \left( \frac{\|\tilde{A}\|}{2} \frac{\eta}{\sqrt{\alpha}} + \frac{p\alpha}{p\alpha + 1} \right) \|y\|,$$

из которой следует, что при  $\alpha(\eta) = O(\eta^2)$  имеет место асимптотика:  $\|\Delta y_\alpha\| \rightarrow 0$  при  $\eta \rightarrow 0$ , т.е. алгоритм, порожденный способом моделирования (вычислительных экспериментов), является *регуляризирующим*.

При этом может вызвать затруднение определение  $p = \|\tilde{A}^+\|^2$ . Оценка  $p$  может быть получена *путем подбора* такого значения  $p$ , при котором огибающая кривая  $\varepsilon(\alpha)$  касается набора кривых  $\sigma_{\text{rel}}(\alpha)$  (см. далее рис. 3.8). Впрочем, при решении примеров  $Q$  и  $P$  не обязательно рассчитывать огибающую  $\varepsilon(\alpha)$ , поскольку она была введена, главным образом, для обоснования способа моделирования. Более подробно технология формирования и решения модельных примеров, составляющая основу способа моделирования (вычислительных экспериментов), изложена в работах [22, с. 246–249], [23, 77, 99, 100].

Добавим, что для повышения эффективности данного способа модельные примеры  $Q$  должны содержать *дополнительную*, априорную, *информацию* о исходном примере  $P$ . В случае решения *обратной задачи спектроскопии* нужно на основе анализа функции  $\tilde{f}_P$  сделать предварительную оценку количества спектральных линий (максимумов) в искомом спектре  $y_P$ , их интенсивностей и частот. Использование такой информации в модельных примерах  $Q$  позволит более точно оценить  $\alpha$ , а также снизить погрешность восстановления спектра. С учетом такой подстройки под реальную ситуацию данный способ следует считать *адаптивным способом*.

**Численная иллюстрация с использованием способа модельных примеров.** Были решены *два иллюстративных примера* [23, 110]. Оба примера описываются интегральным уравнением типа (3.4), причем ИУ (3.4) решается методом конечных сумм (квадратур) с регуляризацией Тихонова с помощью  $m$ -функции `Tikh.m`.

Первый пример — это *исходный пример P (оригинал)*, у которого известны лишь *измеренный спектр*  $\tilde{y}(\nu)$  на сетке узлов  $\nu = c, c + h, \dots, d$ , причем  $c = 0.5$ ,  $d = 10.5$ ,  $h = \text{const} = 0.05$  (в условных единицах), и *аппаратная функция* (АФ):

$$K(\nu, \nu') = \sqrt{\frac{q}{\pi}} \exp \left[ -\frac{q(\nu - \nu')^2}{1 - 0.2(\nu/10)^2} \right], \quad (3.15)$$

где  $q = 15$  (ширина  $K$  убывает с ростом частоты  $\nu$ ). При этом истинный спектр  $z(\nu)$  неизвестен. На рис. 3.6 представлен измеренный спектр  $u(\nu)$  (без шума) и  $\tilde{u}(\nu)$  (с шумом), а также АФ  $K(2, \nu')$  исходного примера.

Из рис. 3.6 видно, что измеренный спектр  $u(\nu)$  имеет сложную структуру, а именно, содержит 6 флуктуаций, две из которых (при  $\nu \approx 2$  и 6.8) скорее всего состоят каждая из двух линий, но они не разрешились из-за того, что АФ имеет некоторую ширину и, тем самым, ограничивает разрешающую способность спектрометра. Таким образом, все указывает на то, что на самом деле в спектре имеется восемь или более спектральных линий. Поэтому в качестве второго примера (*модельного примера*)

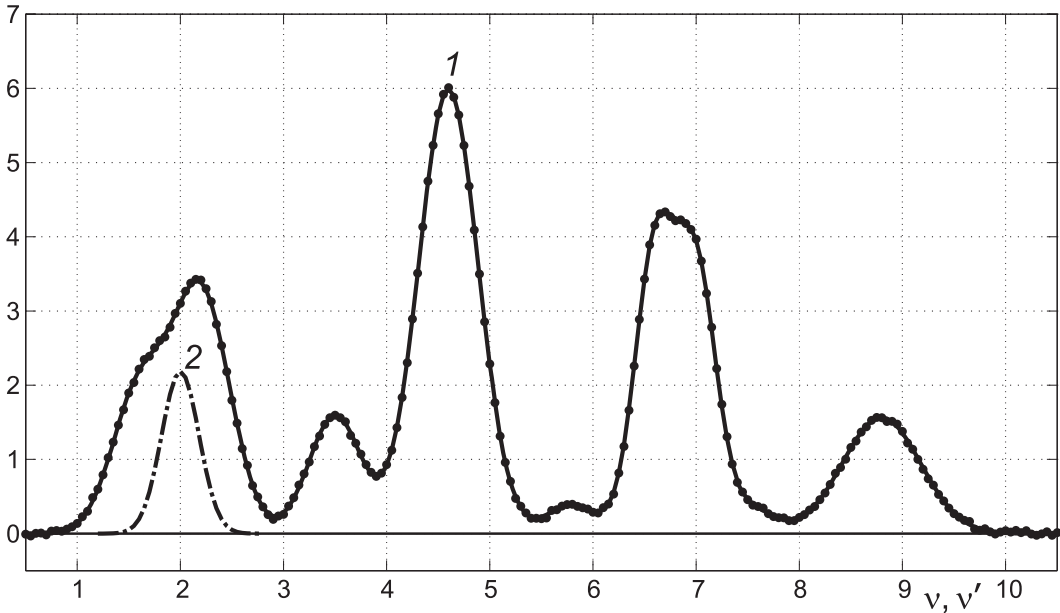


Рис. 3.6. Исходный пример (оригинал)  $P$ :  
 1 — измеренный спектр  $u(\nu)$  и  $\tilde{u}(\nu)$  (с 1%-ным шумом),  
 2 — АФ спектрометра  $K(2, \nu')$

$Q$  рассмотрим близкий к  $P$  пример, у которого истинный спектр  $z(\nu)$  состоит из 10 спектральных линий в виде гауссиан:

$$\begin{aligned}
 z(\nu) = & 3.0 \exp \left\{ -[(\nu - 1.7)/0.25]^2 \right\} + 4.5 \exp \left\{ -[(\nu - 2.3)/0.35]^2 \right\} + \\
 & + 2.5 \exp \left\{ -[(\nu - 3.4)/0.20]^2 \right\} + 0.4 \exp \left\{ -[(\nu - 3.9)/0.10]^2 \right\} + \\
 & + 8.2 \exp \left\{ -[(\nu - 4.7)/0.30]^2 \right\} + 0.5 \exp \left\{ -[(\nu - 5.9)/0.25]^2 \right\} + \\
 & + 9.2 \exp \left\{ -[(\nu - 6.5)/0.10]^2 \right\} + 8.5 \exp \left\{ -[(\nu - 6.9)/0.10]^2 \right\} + \\
 & + 0.4 \exp \left\{ -[(\nu - 7.4)/0.25]^2 \right\} + 2.0 \exp \left\{ -[(\nu - 8.7)/0.50]^2 \right\} .
 \end{aligned} \tag{3.16}$$

Измеренный спектр  $u(\nu)$  в примере  $Q$  был рассчитан по формуле

$$u(\nu) = \int_a^b K(\nu, \nu') z(\nu') d\nu', \quad c \leq \nu \leq d, \tag{3.17}$$

численно путем замены интеграла при каждом  $\nu$  конечной суммой по формуле трапеций с мелким шагом. При этом  $a = 1$ ,  $b = 10$ ,  $c = 0.5$ ,  $d = 10.5$  (в условных единицах),  $h = 0.05$  — шаг дискретизации по  $\nu$ ,  $h/10$  — мелкий шаг по  $\nu'$  при расчете  $u(\nu)$ . К значениям  $u_i$  были добавлены случайные нормальные погрешности с СКО = 0.02 (это

соответствует относительной погрешности  $\delta_{\text{отн}} \approx 1\%$  и СКО = 0.002 ( $\delta_{\text{отн}} \approx 0.1\%$ ). АФ спектрометра в примере  $Q$  была взята в виде (3.15), причем (чтобы промоделировать неточность знания АФ) параметр  $q$  был взят равным  $q = 15$  (это соответствует относительной погрешности АФ  $\xi_{\text{отн}} = 0$ ),  $q = 15 \cdot (1 - 0.001)$  ( $\xi_{\text{отн}} = 0.1\%$ ) и  $q = 15 \cdot (1 - 0.03)$  ( $\xi_{\text{отн}} = 3\%$ ).

Ниже приведен сокращенный вариант текста первой части головной программы NumExpQ.m расчета спектров  $z(\nu)$ ,  $u(\nu)$  и  $\tilde{u}(\nu)$  в модельном примере  $Q$  (прямая задача).

```
%NumExpQ.m (первая часть)
a=1; b=10; n=181; %n - число узлов по nu'
c=0.5; d=10.5; l=201; %l - число узлов по nu
h = (b-a)/(n-1); %h=hnu'=hnu=0.05 - шаг дискретизации
h10 = h/10; %мелкий шаг h10=h/10=0.005
nu10 = a:h10:b; %сетка по nu' с мелким шагом h10
nu = c:h:d; %сетка по nu с шагом h
global q Q %глобальные параметры
q=15; Q=sqrt(q/pi); %для K(nu,nu')
znu1 = ztQ(nu10); %точный спектр z(nu') на частой сетке
u = zeros(1,l); %задание размеров u (для ускорения)
for i=1:l
    kz = Kern(nu(i),nu10).*znu1; %подынтегральная функция
    u(i) = h10*trapz(kz); %измеренный спектр u(nu)
end %i
randn('state',0); %инициализация генератора случайных чисел
u1 = u+0.02*randn(1,l); %добавление 1%-ного шума
u2 = u+0.002*randn(1,l); %добавление 0.1%-ного шума
%Сетка узлов по nu' и точное решение на ней:
nu1 = a:h:b; %сетка узлов по nu'
ze = ztQ(nu1); % точный спектр z(nu') на редкой сетке
%АФ - аппаратная функция спектрометра K(2,nu'):
nu2 = 1.2:h:2.8; %сетка узлов по nu' для K(2,nu')
K2 = Kern(2,nu2); %АФ K(2,nu')
%2D-графика:
figure; %Fig3_7
plot(nu1,ze,'-k', nu,u,'--k', nu,zeros(1,l),'-k');
grid on; axis([.5,10.5,-.5,10]);
```

На рис. 3.7 представлены истинный (точный) спектр  $z(\nu)$  и измеренный (экспериментальный) спектр  $u(\nu)$  в примере  $Q$ .

Далее модельный пример  $Q$  был решен методом квадратур с регуляризацией Тихонова с помощью  $m$ -функции Tikh.m для ряда значений параметра регуляризации  $\alpha$  и была построена зависимость относительной погрешности регуляризованного реше-

ния  $z_\alpha(\nu)$  по отношению к точному решению  $z(\nu)$

$$\sigma_{\text{rel}}(\alpha) = \frac{\|z_\alpha(\nu) - z(\nu)\|}{\|z(\nu)\|}. \quad (3.18)$$

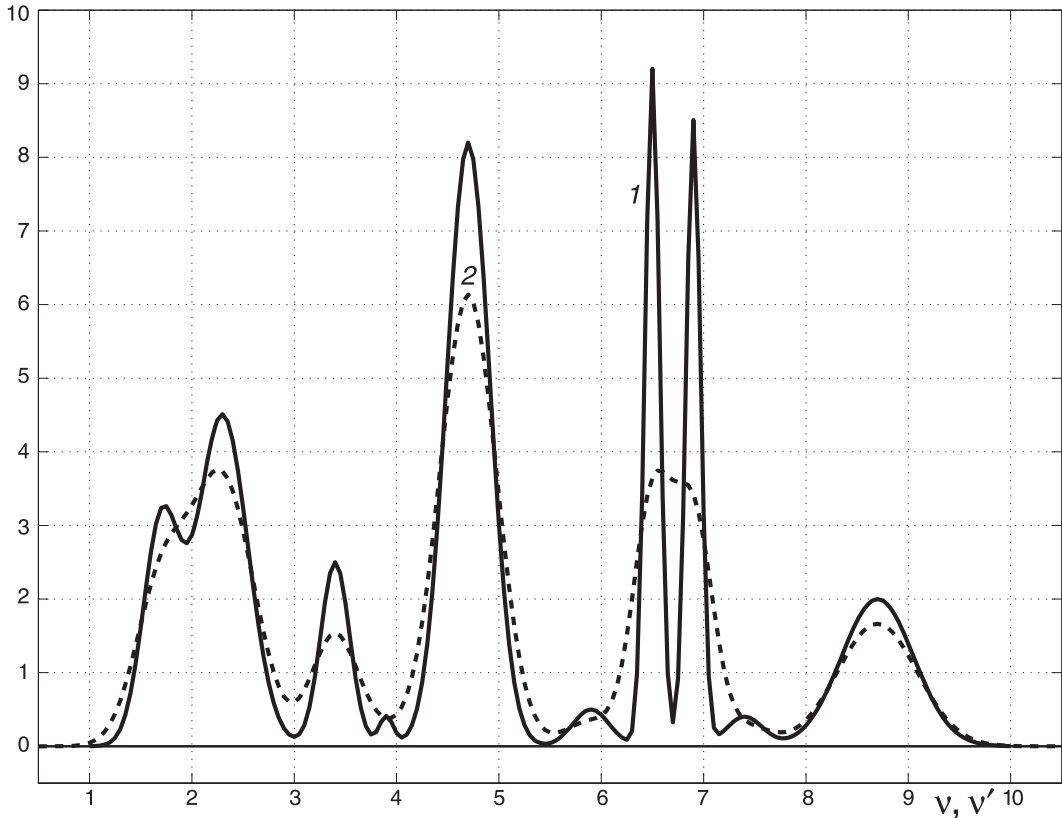


Рис. 3.7. Пример  $Q$  — модельный:  
1 — точный спектр  $z(\nu)$ , 2 — измеренный спектр  $u(\nu)$

Для расчета  $\sigma_{\text{rel}}(\alpha)$  согласно (3.18) была разработана m-функция `error_rel.m`:

```
function s = error_rel(z,ze) %z - некоторое решение, ze - точное решение
s = sqrt(sum((z-ze).^2)/sum(ze.^2)); %отношение норм в пространстве L2
```

Ниже приведен сокращенный вариант текста второй части головной программы `NumExpQ.m` (обратная задача), в которой выполняется расчет функции  $\sigma_{\text{rel}}(\alpha)$  (3.18) для трех вариантов погрешностей  $\delta_{\text{отн}}$  и  $\xi_{\text{отн}}$  и решение одного из вариантов методом квадратур с регуляризацией Тихонова при  $\alpha = \alpha_{\text{opt}}$ .

```
%NumExpQ.m (вторая часть)
alpha = 10.^(-10:.25:2); %значения параметра alpha
```

```

m = length(alpha); %длина массива alpha
sig_rel1 = zeros(1,m); sig_rel2 = zeros(1,m); sig_rel3 = zeros(1,m);
for i=1:m %цикл по alpha
    q=15; Q=sqrt(q/pi); %1) delta_отн=1%, xi=0:
    [z_alpha1,z0_alpha1] = Tikh(u1,nu,nu1,@Kern,alpha(i));
    sig_rel1(i) = error_rel(z0_alpha1,ze);
    q=15*(1-0.001); Q=sqrt(q/pi); %2) delta_отн=xi_отн=0.1%:
    [z_alpha2,z0_alpha2] = Tikh(u2,nu,nu1,@Kern,alpha(i));
    sig_rel2(i) = error_rel(z0_alpha2,ze);
    q=15*(1-0.03); Q=sqrt(q/pi); %3) delta_отн=1%, xi_отн=3%:
    [z_alpha3,z0_alpha3] = Tikh(u1,nu,nu1,@Kern,alpha(i));
    sig_rel3(i) = error_rel(z0_alpha3,ze);
end %i
normA = sqrt(sum(abs(u1).^2)/sum(abs(ze).^2)); %=0.8234
eta = 1e-2; %сумма отн. погрешностей
p = 80; %найденно подбором
ne2 = normA*eta/2; %=0.0182
%В цикле по alpha:
eps = ne2./sqrt(alpha)+p*alpha./(p*alpha+1); %массив
%2D-графика:
lga = log10(alpha); %массив lg(alpha_i)
figure; %Fig3_8
plot(lga,sig_rel1,'-k',lga,sig_rel2,'-k',lga,sig_rel3,'-k',...
     lga,eps,'--k','LineWidth',2.5);
grid on, axis([-10,2,0,1])
alpha0=1e-3; %параметр регуляризации alpha_opt (взят с завышением)
q=15; Q=sqrt(q/pi); %1) delta_отн=1%, xi=0
[z_alpha,z0_alpha] = Tikh(u1,nu,nu1,@Kern,alpha0);
%2D-графика:
figure; %Fig3_9
plot(nu1,ze,'-k', nu1,z0_alpha,'.k', nu,u,'--k', nu1,zeros(1,n));
grid on; axis([1,10,-.5,10]);

```

На рис. 3.8 представлены зависимости  $\sigma_{\text{rel}}(\alpha)$  для нескольких вариантов погрешностей  $\delta_{\text{отн}}$  — относительной погрешности  $u$  и  $\xi_{\text{отн}}$  — относительной погрешности  $K$  (рассмотрено несколько вариантов погрешностей, поскольку в примере-оригинале  $P$  обычно погрешности известны неточно).

На рис. 3.8 представлена также огибающая  $\varepsilon(\alpha)$ , при построении которой было положено:  $\eta = 10^{-2}$ ,  $\|A\| = \|u\|_{L_2}/\|z\|_{L_2} = 0.823$ , а значение  $p$  было подобрано так, чтобы огибающая кривая  $\varepsilon(\alpha)$  касалась набора кривых  $\sigma_{\text{rel}}(\alpha)$ . Из рис. 3.8 видно, что оптимальное значение параметра регуляризации  $\alpha$ , например, для варианта 1 по минимуму кривой  $\sigma_{\text{rel}}(\alpha)$  равно приблизительно  $10^{-3.7}$ . Однако в примере  $Q$  при таком значении  $\alpha$  в решении  $z_\alpha(\nu)$  хорошо восстанавливаются сильные линии, а слабые ли-

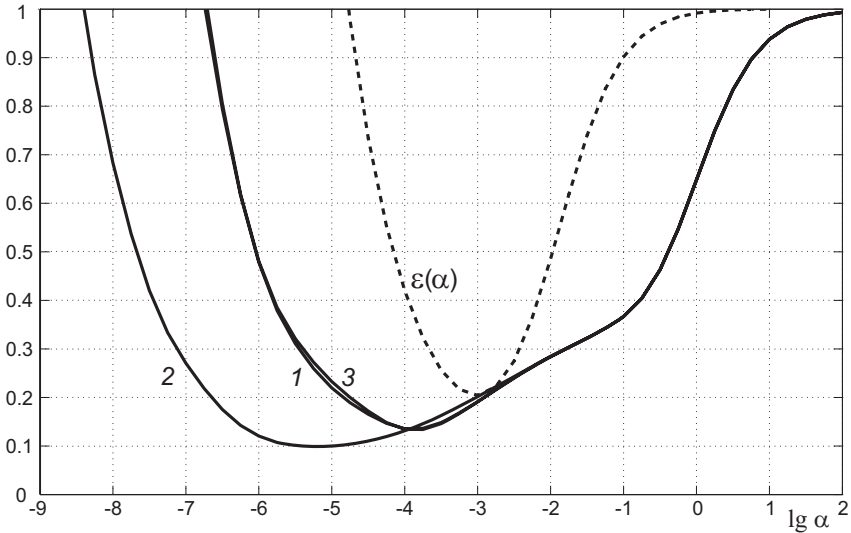


Рис. 3.8. Пример  $Q$  (модельный).

Относительная погрешность  $\sigma_{\text{rel}}(\alpha)$  решения  $z_\alpha(\nu)$ :

- 1 — при  $\delta_{\text{отн}} \approx 1\%$ ,  $\xi_{\text{отн}} = 0$ , 2 — при  $\delta_{\text{отн}} \approx \xi_{\text{отн}} \approx 0.1\%$ ,
- 3 — при  $\delta_{\text{отн}} \approx 1\%$ ,  $\xi_{\text{отн}} \approx 3\%$ ;  $\varepsilon(\alpha)$  — огибающая ( $p = 80$ )

нии отягощены ложными флуктуациями и лучше восстанавливаются при большем значении  $\alpha$ . На рис. 3.9 представлен истинный спектр  $z(\nu)$  и восстановленный спектр  $z_\alpha(\nu)$  при несколько завышенном  $\alpha$ , а именно, при  $\alpha = 10^{-3}$ . Кроме того, кривая  $\varepsilon(\alpha)$  имеет минимум также при  $\alpha = 10^{-3}$ . Видим, что спектр в модельном примере  $Q$  восстановился вполне удовлетворительно: разрешились близкие линии и восстановились слабые линии, лишь на краю области (при  $\nu \gtrsim 9.5$ ) наблюдаются слабые ложные флуктуации в спектре  $z_\alpha(\nu)$  — знакомый нам по гл. 2 краевой эффект Гиббса.

При значении  $\alpha = 10^{-3}$ , выбранном в модельном примере  $Q$ , восстановлен спектр также в примере–оригинале  $P$  (см. рис. 3.10). Как и в примере  $Q$ , в примере  $P$  разрешились близкие линии и восстановились слабые линии, но также проявился краевой эффект Гиббса (ложные флуктуации при  $\nu \gtrsim 9.5$ ), причем не только на краю, но и в середине спектра: слабая линия при  $\nu \approx 5.8$  дала ложное расщепление на две линии. Однако это слабые эффекты — эффекты на уровне погрешностей метода.

Практическое использование изложенной методики может привести к повышению разрешающей способности спектрометра.

### Контрольные задания и вопросы

1. Дайте определение аппаратной функции (АФ) спектрометра. Проиллюстрируйте определение рисунком и приведите пример АФ в виде формулы.
2. В чем состоят отличия экспериментального спектра от истинного?
3. Сформулируйте задачу определения истинного спектра по экспериментальному спектру и АФ спектрометра.
4. Обоснуйте вывод основного (интегрального) уравнения в задаче восстановления



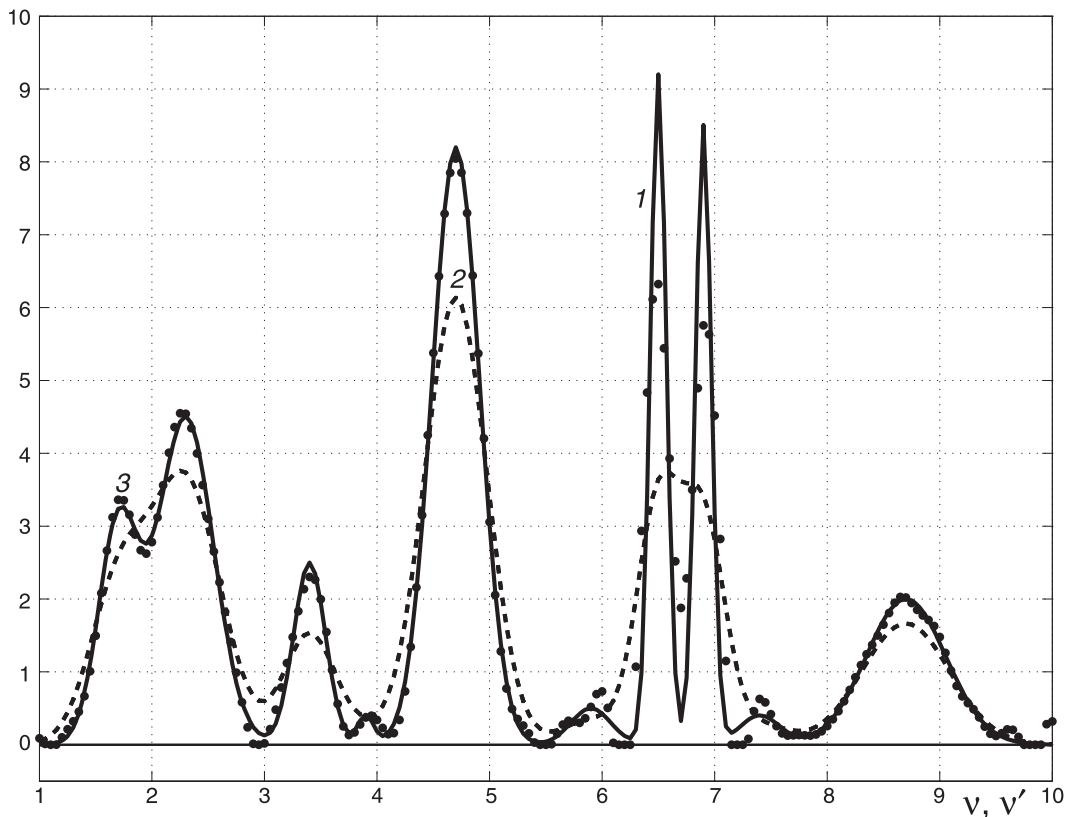


Рис. 3.9. Пример  $Q$  (модельный):

1 — истинный спектр  $z(\nu)$ , 2 — измеренный спектр  $u(\nu)$ ,  
 3 — восстановленный спектр  $z_\alpha(\nu)$  при  $\delta_{\text{отн}} \approx 1\%$ ,  $\xi_{\text{отн}} = 0$ ,  $\alpha = 10^{-3}$

непрерывного спектра.

5. Изложите упрощенный вариант решения ИУ Фредгольма I рода методом квадратур с регуляризацией Тихонова (задание повышенной трудности).

6. Кратко изложите способ моделирования восстановления спектра. В чем сходства и различия исходного  $P$  и модельного  $Q$  примеров?

### 3.3. Задача спектроскопии в случае дискретного спектра

**Математическое описание задачи.** Теперь рассмотрим случай *дискретного (линейчатого) спектра*, когда искомый спектр  $z(\nu)$  состоит из отдельных дискретных (монохроматических) линий, характеризующихся их частотами и амплитудами [64], [102, с. 85–90] (см. рис. 3.11). Такой спектр имеют, например, рассеянные межзвездные туманности и низкотемпературная, в частности, газоразрядная плазма.

В случае дискретного спектра измеренное значение интенсивности  $u(\nu)$  при настройке спектрометра на некоторую частоту  $\nu$  равно сумме интенсивностей (амплитуд) всех линий с весовой функцией, равной аппаратной функции (АФ) спектрометра

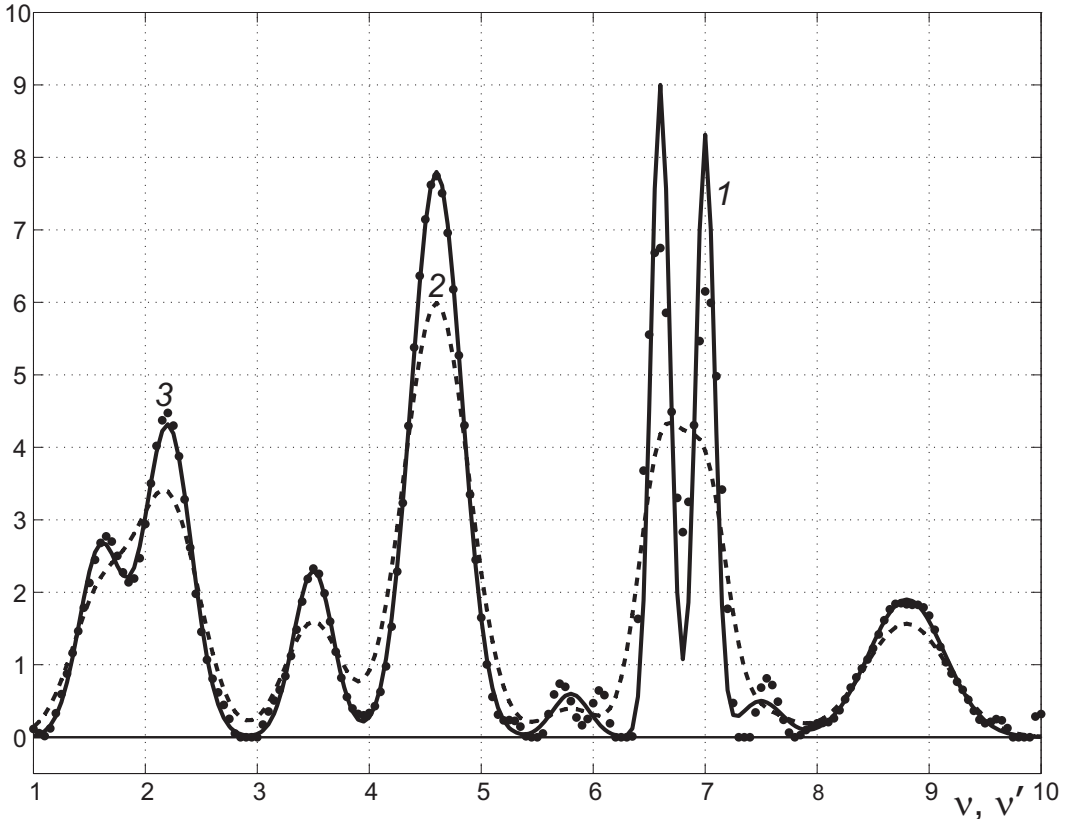


Рис. 3.10. Пример  $P$  (оригинал):

1 — истинный спектр  $z(\nu)$ , 2 — измеренный спектр  $u(\nu)$ ,  
 3 — восстановленный спектр  $z_\alpha(\nu)$  при  $\delta_{\text{отн}} \approx 1\%$ ,  $\xi_{\text{отн}} = 0$ ,  $\alpha = 10^{-3}$

$K$  (см. рис. 3.11), т. е.

$$u(\nu) = \sum_{j=1}^n z_j K(\nu, \nu_j'), \quad (3.19)$$

где  $z_j$  — амплитуда (интенсивность)  $j$ -й линии,  $\nu_j'$  — ее частота,  $n$  — число линий. Настроив спектрометр на различные частоты  $\nu$  и учитывая зашумленность измерений, получим:

$$\sum_{j=1}^n K(\nu_i, \nu_j') z_j + F = \tilde{u}(\nu_i), \quad i = \overline{1, m}, \quad c \leq \nu_i \leq d, \quad (3.20)$$

где  $\nu_i$  — дискретный отсчет,  $m$  — число таких отсчетов,  $[c, d]$  — диапазон настраиваемых частот,  $\tilde{u}(\nu_i) = u(\nu_i) + \delta u(\nu_i)$ ,  $\delta u$  — случайная компонента шума измерений,  $F$  — детерминированная компонента шума (фон) (см. рис. 3.12).

В (3.20) известны (измерены или заданы)  $\tilde{u}(\nu_i)$ ,  $K(\nu_i, \nu_j')$ ,  $\nu_j$ ,  $c$ ,  $d$ ,  $m$ , а искомыми являются  $z_j$ ,  $\nu_j'$ ,  $n$ ,  $F$  — амплитуды и частоты линий, их число, а также детерминированная компонента шума. Соотношение (3.20) есть *система линейно-нелинейных*

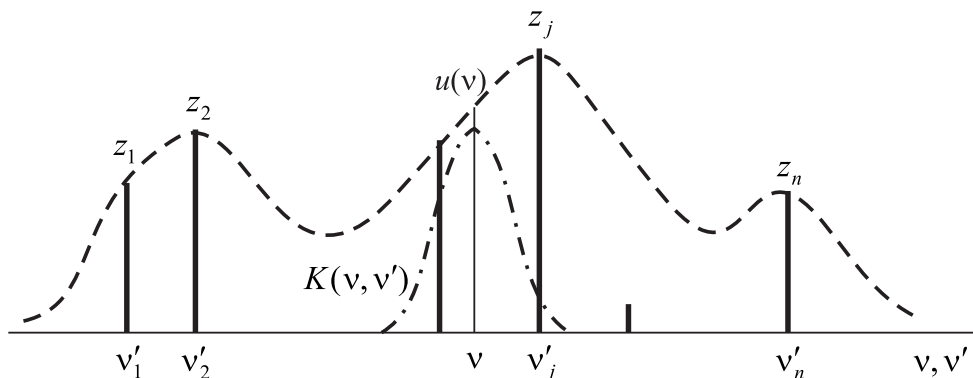


Рис. 3.11. Схема дискретного спектра (без шума)

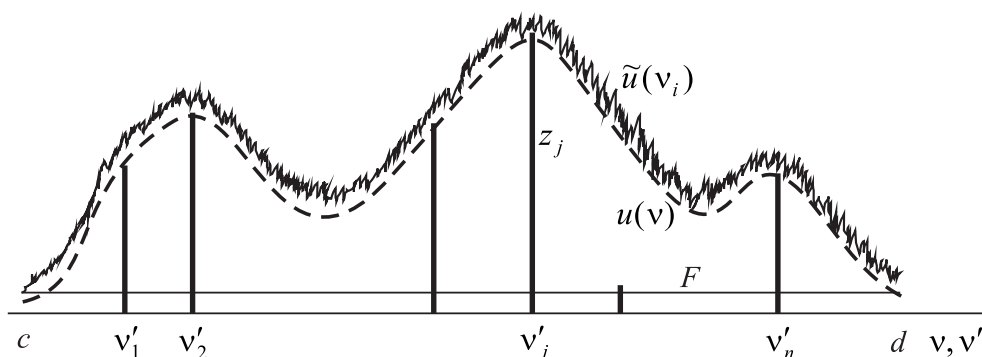


Рис. 3.12. Схема дискретного спектра с шумом

уравнений (СЛНУ), поскольку часть неизвестных ( $z_j$  и  $F$ ) входит линейно, а часть ( $\nu_j'$ ) — нелинейно.

**О решении СЛНУ.** Рассмотрим вопрос о решении СЛНУ (3.20). Система уравнений (3.20) может рассматриваться и как система нелинейных уравнений (СНУ) относительно  $2n$  неизвестных  $z_j$  и  $\nu_j'$ , а также  $F$  при некотором  $n$ . Ее можно решать известными методами решения СНУ, а именно,

- методами решения СНУ без ограничений на решение: методами Ньютона–Канторовича, градиента, хорд и др. [9, 16, 37, 115] или
- методами решения СНУ с ограничениями на решение (методами нелинейного программирования): проекции градиента, оврагов и др. [37, 121, 132].

При этом более эффективным является применение методов нелинейного программирования, так как на неизвестные  $z_j$ ,  $\nu_j'$  и  $F$  можно наложить следующие ограничения (способствующие повышению точности решения):

$$z_j > 0, \quad \nu_j' \in [a, b], \quad F \geq 0, \quad (3.21)$$

где  $[a, b]$  — некоторая область, более узкая, чем  $[c, d]$ .

Однако эти методы не учитывают специфику системы (3.20) (амплитуды  $z_j$  и фон  $F$  входят линейно, а частоты  $\nu_j'$  — нелинейно), кроме того, они оставляют открытым вопрос о числе линий  $n$ . Решение системы (3.20) будет более эффективным (потребуется меньше компьютерного времени и памяти, понизится вероятность появления ложных корней нелинейной системы и т. д.), если учесть специфику системы (3.20).

Для решения СЛНУ типа (3.20) можно использовать, например, метод Прони [53], предназначенный для решения СЛНУ, однако он подходит лишь для СЛНУ с матрицей Вандермонда (когда  $K(\nu_i, \nu_j')$  изменяется вдоль строки по геометрической прогрессии), однако матрица в (3.20) таковой, вообще говоря, не является. Был испробован алгоритм Пиблза–Берковица [84], основанный на представлении АФ рядом Тейлора, но он оказался весьма неточным, а также алгоритм Фальковича–Коновалова [127], но это слишком громоздкий алгоритм. Можно использовать так называемый метод переменных проекций (the variable projection method) [154, 155, 163], в котором также решается СЛНУ типа (3.20), однако в нем для отыскания частот  $\nu_j'$  используется нелинейный метод (типа Гаусса–Ньютона).

Поэтому для решения СЛНУ (3.20) предлагается использовать *алгоритм интегральной аппроксимации* [100], показавший свою эффективность в задаче спектроскопии [64], [102, с. 88–90] и в обработке сигналов [100].

**Алгоритм интегральной аппроксимации решения СЛНУ.** Согласно данному алгоритму,

1. Решается ИУ (3.4) или (3.5) методом регуляризации Тихонова. При этом необходимо, чтобы число дискретных отсчетов  $m$  измеренного спектра было большим (несколько сот), а параметр регуляризации  $\alpha$  имел заниженное значение. Эти меры необходимы для разрешения близких линий. В результате будет получено решение  $z_\alpha(\nu')$  с малым шагом дискретизации  $\Delta\nu'$ , в котором могут разрешиться близкие линии, но (из-за пониженности  $\alpha$ ) возникнут также ложные флуктуации–линии.

2. В решении  $z_\alpha(\nu')$  выделяется ограниченное количество максимумов, а именно,  $L \leq N$  наиболее мощных максимумов, где  $N$  задается на основе дополнительной информации так, чтобы  $N \geq n$ , где  $n$  — предполагаемое число линий. Фиксируются частоты наиболее мощных максимумов  $\tilde{\nu}_j'$ ,  $j = \overline{1, L}$ .

3. Решается уточняющая СЛАУ

$$\sum_{j=1}^L K(\nu_i, \tilde{\nu}_j') \tilde{z}_j + \tilde{F} = \tilde{u}(\nu_i), \quad i = \overline{1, m}, \quad c \leq \nu_i \leq d, \quad (3.22)$$

методом наименьших квадратов Гаусса (без регуляризации) относительно  $L$  амплитуд  $\tilde{z}_j$  и фона  $\tilde{F}$ .

4. Оставляются лишь те  $\tilde{z}_j$  и  $\tilde{F}$ , для которых справедливо:

$$\tilde{z}_j \geq Z, \quad j = \overline{1, k}, \quad \tilde{F} > 0, \quad (3.23)$$

где  $Z > 0$  — некоторый априори заданный барьер, а  $k \leq L$  — количество  $\tilde{z}_j$ , преодолевших барьер  $Z$ . При этом барьер  $Z$  можно определить, используя соотношение [126, с. 126]:

$$Z = \sigma_u \sqrt{-2 \ln F_{\text{лт}}}, \quad (3.24)$$

где  $\sigma_u$  — среднеквадратическое значение случайной составляющей помехи  $\delta u$ , полагаемое известным, а  $F_{\text{лт}} \in [0, 1]$  — задаваемая условная вероятность ложной тревоги. Можно величину барьера  $Z$  не связывать со значением  $F_{\text{лт}}$ , например, когда все ложные максимумы при решении уточняющей СЛАУ принимают отрицательные значения или значения, близкие к нулю.

Достоинством алгоритма интегральной аппроксимации является то, что наиболее сложная часть задачи — определение нелинейно входящих значений частот  $\nu_j'$ , а также  $n$  — их числа — решается линейно, а именно, путем решения линейного ИУ (3.4) или (3.5).

**Численный пример.** Алгоритмом интегральной аппроксимации был решен следующий модельный пример (типа примера 2 из [100, ч. III] и примера из [102, с. 88–90]). Данные примера: истинное число линий  $n = 6$ , их амплитуды (в условных единицах):  $z_1 = 5.3$ ,  $z_2 = 5.1$ ,  $z_3 = 3.3$ ,  $z_4 = 3.7$ ,  $z_5 = 3.9$ ,  $z_6 = 3.8$ , их частоты (также в условных единицах):  $\nu_1' = 2.43$ ,  $\nu_2' = 2.49$ ,  $\nu_3' = 3.07$ ,  $\nu_4' = 3.56$ ,  $\nu_5' = 3.62$ ,  $\nu_6' = 3.68$ . Аппаратная функция (АФ) спектрометра равна

$$K(\nu, \nu') = 0.9 \sqrt{\frac{\nu}{\pi}} \exp\left(-\frac{(\nu - \nu')^2}{2\sigma^2(1 - 0.18\nu)}\right), \quad (3.25)$$

где  $\sigma = 0.05$ . Видим, что ширина  $K$  уменьшается с увеличением частоты настройки спектрометра  $\nu$ .

Однако число линий  $n$ , их амплитуды и частоты неизвестны на практике, а известен измеренный спектр  $z(\nu)$ , а также АФ  $K(\nu, \nu')$ . Пусть измерения  $z(\nu)$  выполнены при  $\nu \in [c, d] = [2, 4]$  и число дискретных отсчетов  $m = 401$ , т.е. через шаг  $h = \Delta\nu = (d - c)/(m - 1) = 0.005$  (в условных единицах). Моделируем измеренный с шумом спектр (ср. (3.20)):

$$\tilde{u}(\nu_i) = \sum_{j=1}^n K(\nu_i, \nu_j') z_j + \delta u + F, \quad i = \overline{1, m}, \quad \nu_i = c + h(i - 1), \quad (3.26)$$

причем  $\sigma_u = 0.05$ ,  $F = 0.1$ . На рис. 3.13 представлены: истинный дискретный (линейчатый) спектр из 6 линий, измеренный (экспериментальный) спектр без шума  $u(\nu)$  и с шумом  $\tilde{u}(\nu)$ , а также АФ спектрометра на низкой и высокой частоте.

Приводим первую часть текста головной программы `AlgIntApp.m`, выполняющей решение прямой задачи в модельном примере.

```
%AlgIntApp.m (первая часть)
z = [5.3, 5.1, 3.3, 3.7, 3.9, 3.8]; %6 точных амплитуд
nu1 = [2.43, 2.49, 3.07, 3.56, 3.62, 3.68]; %6 точных частот
n = length(z); c=2; d=4; m=401; m1=m-1; h=(d-c)/m1; F=0.1;
global s2 %для K2(nu,nu1)
sigma = 0.05; s2 = 2*sigma^2;
%Вычисление измеренного спектра:
nu = c:h:d; %сетка по nu 1x401
u = zeros(1,m); %задание размеров u (для ускорения)
```

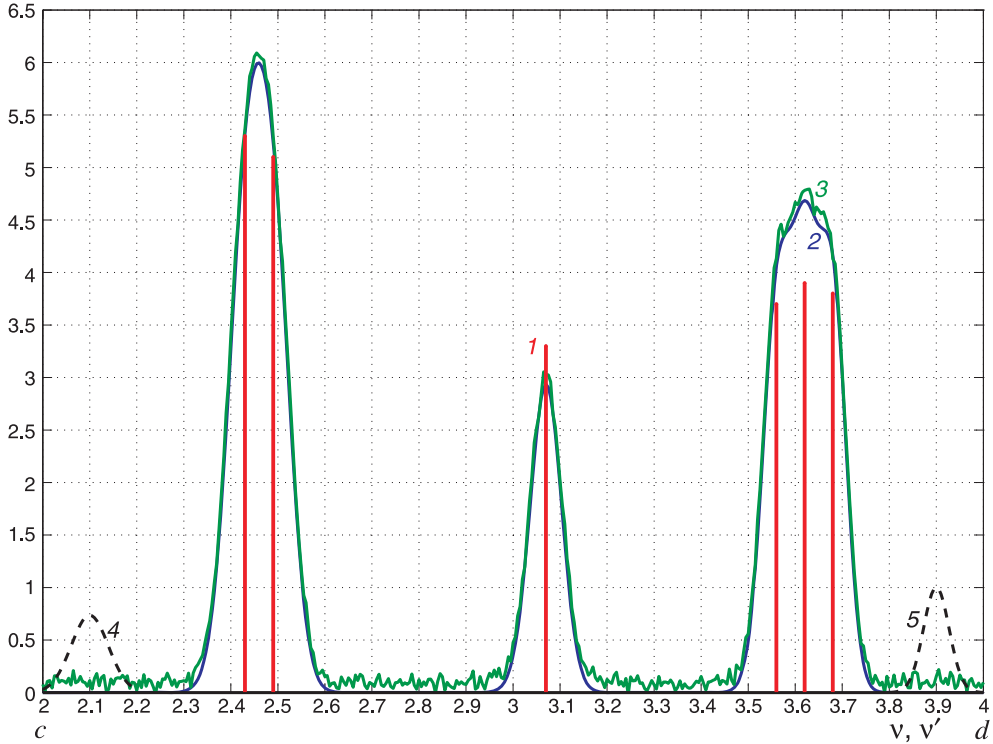


Рис. 3.13. Модельный пример (прямая задача):

- 1 — истинный линейчатый спектр,
- 2 — измеренный спектр  $u(\nu)$  без шума,
- 3 — измеренный зашумленный спектр  $\tilde{u}(\nu)$ ,
- 4 —  $K(2.1, \nu_j')$  — АФ на низкой частоте,
- 5 —  $K(3.9, \nu_j')$  — АФ на высокой частоте

```

for i=1:m
    nui = nu(i); s=0;
    for j=1:n
        s = s+K2(nui,nu1(j))*z(j);
    end %j
    u(i) = s;
end %i
nul = c:h:2.2; K2left = K2(2.1,nul);
nur = 3.8:h:d; K2right = K2(3.9,nur);
randn('state',0);
u1 = u+.05*randn(1,m)+F; %добавление нормального шума
%2D-график:
figure; %Fig3_13

```

```
plot(nu,u,nu,u1,nu,zeros(1,m),'-k',nul,K2left,'--k',...
     nur,K2right,'--k','LineWidth',2), grid on, hold on
stem(nul,z,'-r','MarkerSize',0,'LineWidth',2.5); %вертикальные черты
```

При этом АФ спектрометра  $K(\nu, \nu')$  оформлена в виде  $m$ -функции:

```
function kern = K2(nu,nul) %m-функция
global s2
kern = 0.9*sqrt(nu/pi).*exp(-(nu-nul).^2./(s2*(1-0.18*nu)));
```

Далее из рис. 3.13 видно, что в измеренном спектре близкие линии (две слева и три справа) не разрешились. Наша задача — разрешить их. Для этого воспользуемся алгоритмом интегральной аппроксимации.

Решаем ИУ (3.4) методом квадратур с регуляризацией Тихонова с помощью  $m$ -функции `Tikh.m` (ее текст см. выше). При этом сетки узлов по  $\nu$  и  $\nu'$  полагаем равномерными, причем  $a = c$ ,  $b = d$ . На рис. 3.14 приведено решение  $z_\alpha(\nu')$  при  $\alpha = 10^{-6}$ .

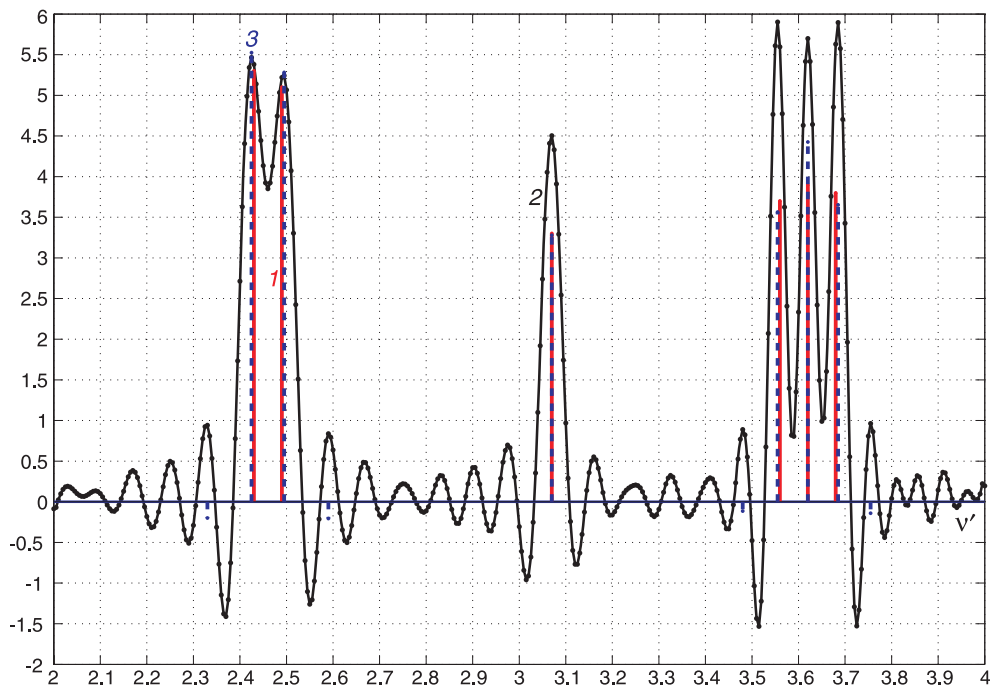


Рис. 3.14. Модельный пример (обратная задача).

Решение алгоритмом интегральной аппроксимации:

- 1 — точные интенсивности линий  $z_j$ ,
- 2 — решение  $z_\alpha(\nu')$  ИУ (3.4) методом квадратур с регуляризацией Тихонова при  $\alpha = 10^{-6}$  (отображено  $0.05z_\alpha(\nu')$ ),
- 3 — значения  $\tilde{z}_j$ , полученные путем решения уточняющей СЛАУ

Из рис. 3.14 видно, что разрешились все истинные спектральные линии, однако появилось много ложных линий (максимумов в решении  $z_\alpha(\nu')$ ). В решении  $z_\alpha(\nu')$  берем первые  $L = 10$  наиболее мощных максимумов. Их частоты:  $\tilde{\nu}'_j = 2.33, 2.425, 2.495, 2.59, 3.07, 3.48, 3.555, 3.62, 3.685, 3.755$ . Решаем уточняющую систему  $m = 401$  линейных алгебраических уравнений (3.22) относительно  $L + 1 = 11$  неизвестных: 10 амплитуд  $\tilde{z}_j$  и фона  $F$  методом наименьших квадратов Гаусса. В результате решения уточняющей СЛАУ получено:  $\tilde{z}_j = -0.1989, 5.5245, 5.2809, -0.2051, 3.2792, -0.1149, 3.5633, 4.4262, 3.6520, -0.1412$ , а  $\tilde{F} = 0.0991$ . На рис. 3.14 отмечены полученные значения  $\tilde{z}_j$ . Видим, что все ложные максимумы получили отрицательные значения  $\tilde{z}_j$ , а истинные максимумы получили значения  $\tilde{z}_j$ , весьма близкие к точным значениям амплитуд  $z_j$ . В этом случае нет необходимости связывать барьер  $Z$  со значением  $F_{\text{лт}}$ , другими словами, нужно положить  $Z = 0$ .

В результате можно констатировать, что в данном примере все шесть спектральных линий разрешились и с приемлемой точностью определились их частоты и интенсивности, а также фон  $F$ , причем ни одна линия не потерялась и ни одна ложная линия не появилась, хотя помехо-сигнальная ситуация в этом примере выбрана специально сложная (см. рис. 3.13), чтобы продемонстрировать возможности алгоритма интегральной аппроксимации.

В заключении приведем вторую часть головной программы `AlgIntApp.m`, где выполнялось решение модельного примера алгоритмом интегральной аппроксимации.

```
%AlgIntApp.m (вторая часть)
%Решение ИУ (3.4) методом квадратур с регуляризацией Тихонова:
alpha = 1e-6; %параметр регуляризации (заниженный)
z_alpha = Tikh(u1,nu,nu,@K2,alpha); %z_alpha 1x401
%2D-график:
figure; %Fig3_14
plot(nu,0.05*z_alpha,'-k', nu,0.05*z_alpha,'.k',...
      nu,zeros(1,m),'-b','LineWidth',1.7,'MarkerSize',10)
grid on, hold on
stem(nu1,z,'-r','MarkerSize',0,'LineWidth',2.5)
%Из решения ИУ берем L=10 частот наиболее мощных максимумов:
nut1 = [2.330, 2.425, 2.495, 2.590, 3.070, ...
        3.480, 3.555, 3.620, 3.685, 3.755];
L = length(nut1); %=10
L1=L+1; %=11
%Система (3.22) m линейных алгебраических уравнений относительно
%L1 неизвестных (10 наиболее мощных амплитуд и фона F):
A = zeros(m,L1); %задание размеров матрицы A (для ускорения)
for i=1:m
    nui = nu(i); A(i,L1) = 1;
    for j=1:L
        A(i,j) = K2(nui,nut1(j)); %элементы матрицы СЛАУ 401x11
    end %j
```



```

end %i
%Решение СЛАУ (3.22) МНК Гаусса относительно
%L амплитуд zt и фона F (без регуляризации):
ztF = ((A'*A)\(A'*u1'))'; %ztF 1x11
zt = ztF(1:L); %амплитуды 1x10:
%-0.1989 5.5245 5.2809 -0.2051 3.2792
%-0.1149 3.5633 4.4262 3.6520 -0.1412
Ft = ztF(L1); %фон =0.0991
hold on
stem(nut1,zt,'--b','MarkerSize',0,'LineWidth',2.5)

```

**О некоторых особенностях программирования в системе MatLab.** MatLab — это система программирования, имеющая ряд специфических особенностей, отличающих ее от таких языков программирования, как Fortran, C/C++, Pascal и т. д. И если программист ранее работал, например, на C++, а теперь перешел на MatLab и навыки работы с C++ перенес на MatLab, то его программы в системе MatLab будут, скорее всего, весьма неоптимальными, в первую очередь, медленно работающими.

Отметим некоторые полезные приемы, которые могут повысить скорость выполнения головных программ и m-функций в системе MatLab.

1) В системе MatLab нет явного описания переменных, в частности, массивов, поэтому компилятор часто заранее "не знает", сколько ячеек нужно выделить под массив, и обычно делает это итерационно (последовательно), что приводит к снижению скорости. Кроме того, может иметь место *фрагментация памяти* при работе с большими массивами. Чтобы этого избежать, нужно перед работой с массивом выделить под него нужное количество ячеек, лучше всего, с помощью функции `zeros`, например:

```

u = zeros(1,1);
for i=1:l
    u(i) = ...
end %i

```

2) В системе MatLab максимально оптимизированы операции над матрицами и векторами. При этом надо стараться не пользоваться оператором `for`, а использовать оператор двоеточие, например, вместо

```

for i=1:m
    x(i) = c+h*(i-1);
end %i

```

следует писать:

```
x = c:h:d;
```

или вместо

```

for i=1:mg
    om1(i) = dom1*(i-1);
end %i

```

следует писать (см. `refocusingT.m`):

```
om1 = dom1*(0:mg-1);
```

3) Надо стараться не пользоваться оператором `if`, например, вместо (см. (3.11))

```
for j=1:n
  if (j==1) r(1) = (s(2)-s(1))/2;
  elseif (j<n) r(j) = (s(j+1)-s(j-1))/2;
  else r(n) = (s(n)-s(n-1))/2;
  end %if
end %j
```

следует писать:

```
r(1) = (s(2)-s(1))/2; r(n) = (s(n)-s(n-1))/2;
for j=2:n-1
  r(j) = (s(j+1)-s(j-1))/2;
end %j
```

или еще эффективнее (см. `Tikh.m`):

```
r = [s(2)-s(1), s(3:n)-s(1:n-2), s(n)-s(n-1)]/2;
```

4) Надо чаще использовать поэлементные операции (при работе с векторами или матрицами), например (см. `NumExpQ.m`):

```
alpha = 10.^(-10:0.25:2); %alpha = 1e-10, ..., 1e2
```

или (см. (3.18) и `error_rel.m`)

```
function s = error_rel(z,ze)
s = sqrt(sum((z-ze).^2)/sum(ze.^2));
```

или (см. `Kern.m`)

```
function k = Kern(nu,nu1)
global q Q
k = Q*exp(-q*(nu-nu1).^2./(1-0.2*(nu/10).^2));
```

5) Несмотря на то, что надо стремиться не использовать оператор `for`, иногда трудно отказаться от его применения. Такая ситуация может возникнуть, когда встречается вложенный цикл, тем более, с использованием функции (см. `NumExpP.m`):

```
for i=1:l
  kz = Kern(nu(i),nu10).*znu1; %подынтегральная функция
  u(i) = h10*trapz(kz); %спектр u(nu) 1x1
end %i
```

Тем не менее, и в 2-мерном случае можно полностью избежать использования оператора `for`, если, например, использовать функцию `repmat` (см. `Tikh.m`): вместо

```

A = zeros(m,n); %выделяем память под матрицу A mхn
for i=1:m
    xi = x(i);
    for j=1:n
        A(i,j) = r(j)*kern(xi,s(j));
    end %j
end %i

```

можно использовать всего лишь один (!) оператор:

```
A = repmat(r,m,1).*kern(repmat(x',1,n),repmat(s,m,1)); %A mхn
```

Такой оператор помимо компактности, повышает скорость. Однако он приводит к излишней загрузке памяти, так как 1-мерные массивы  $\mathbf{r}$ ,  $\mathbf{x}$  и  $\mathbf{s}$  порождают 2-мерные массивы, например, `repmat(r,m,1)` порождает матрицу  $m \times n$ , в которой  $m$  раз продублирована вектор-строка  $\mathbf{r}$   $1 \times n$ .

В 2-мерном случае можно использовать также *векторизацию циклов* [36, с. 70–73]. Например, выделение (вырезку) части изображения путем векторизации можно выполнить следующим образом (см. `RecDefImTt.m`): `W_Kr = W_K(1+r:M-r,1+r:N-r)`; и это будет выполнено значительно быстрее, чем с помощью двойного цикла `for`. Векторизацию циклов можно реализовать также с помощью внешней  $m$ -функции `meshgrid` [36, с. 70–72], которая, как и функция `repmat`, из 1-мерных массивов формирует 2-мерные за счет многократного копирования строк и столбцов.

Еще одним примером компактной записи может быть запись решения (3.14) в виде одного оператора (см. `Tikh.m`): `y_alpha = ((alpha*eye(n)+A'*A)\(A'*f'))'`;

6) MatLab наиболее эффективен, когда он оперирует исключительно с матрицами и его работа может сильно замедляться, когда в вычислениях наряду с матрицами участвуют функции. Выше уже было отмечено, что вычисления в рамках системы MatLab по схеме (2.92) с использованием массивов  $w$  и  $g$  и функции  $h$  выполняются на 2 порядка (!) медленнее, чем по схеме (2.93), в которой вместо функции  $h$  введена соответствующая ей матрица  $M$  (см. `refocusingF.m`).

**Критерий Рэлея и редукционная проблема Рэлея.** Решение задач реконструкции искаженных изображений и восстановления спектров еще раз продемонстрировало, что при выполнении достаточно эффективной математической обработки изображений и спектров в принципе не ставят пределов на разрешение ни критерий Рэлея, ни дифракционные ограничения [22, с. 226]. Поясним это.

Рассмотрим два точечных источника света одинаковой интенсивности [128, с. 615], например, две звезды, наблюдаемые в телескоп (или две дискретные линии, измеряемые спектрометром). Вследствие дифракции света даже при отсутствии различного рода погрешностей (аббераций, ошибок изготовления оптической системы, шумов и т.д.) оптическая система воспроизводит каждый точечный источник (математически  $\delta$ -функцию) в монохроматическом свете в виде светлого пятна, окруженного попеременно темными и светлыми кольцами (см. рис. 3.15).

Рэлей (1879) сформулировал следующий критерий [37, с. 152–153], [67, с. 196–198], [94], [128, с. 615, 651].

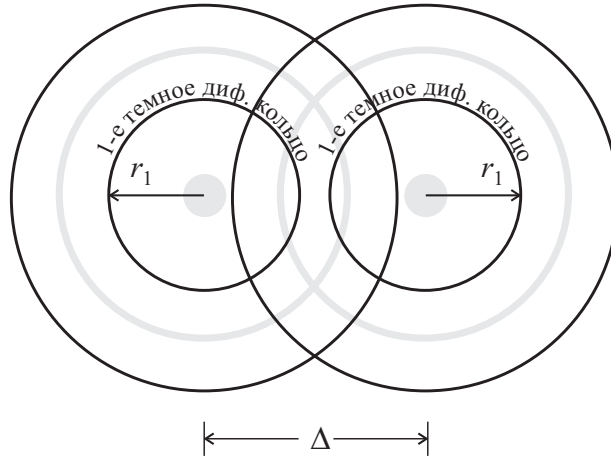


Рис. 3.15. Два точечных источника с дифракционными кольцами.

$\Delta$  — расстояние между источниками,  
 $r_1$  — радиус первого темного дифракционного кольца

Изображения двух источников можно наблюдать отдельно (без математической обработки), если расстояние между ними (между центрами дифракционных пятен)  $\Delta$  не меньше радиуса  $r_1$  первого темного дифракционного кольца (радиуса пятна Эйри). На рис. 3.16 изображен предельный случай, когда  $\Delta = r_1$ , т. е. когда центр дифракционного пятна каждого источника пересекается с краем первого темного кольца другого. Видим, что между источниками имеет место небольшой (примерно 20%-ный) провал интенсивности, позволяющий наблюдать источники отдельно.

Критерий Рэлея можно сформулировать иначе [37, с. 152]. Пусть на входе измерительной системы (телескопа, спектрометра и т.д.) есть два точечных источника (две звезды, две спектральные линии и т.д.) одинаковой интенсивности на расстоянии  $d$  и пусть размытие суммарной интенсивности  $I(r)$  (см. рис. 3.16) вызвано не дифракцией света, а аппаратной функцией системы. Тогда, согласно критерию Рэлея, источники различимы, если провал между ними равен примерно 20%. В этом случае  $d = \Delta$ .

Исходя из критерия Рэлея, была получена следующая формула для минимального углового расстояния между источниками (в рад), при котором они еще могут наблюдаться отдельно в оптическую систему (телескоп) [128, с. 615]:

$$\Delta_{\min} = 1.21 \lambda / D, \text{ рад}, \quad (3.27)$$

где  $\lambda$  — длина волны света,  $D$  — диаметр входного зрачка (апертуры) оптической системы ( $\lambda$  и  $D$  — в одинаковых единицах, например, в мм).

Выведена еще более конкретная формула, дающая предел разрешения телескопа при визуальном наблюдении в него [128, с. 615]:

$$\delta = 140 / D, \quad (3.28)$$

где  $\delta$  — в угловых секундах,  $D$  — в мм. Формула (3.28) получена из формулы (3.27) при  $\lambda = 560 \text{ нм} = 0.56 \cdot 10^{-3} \text{ мм}$ , что соответствует средней длине волны света,

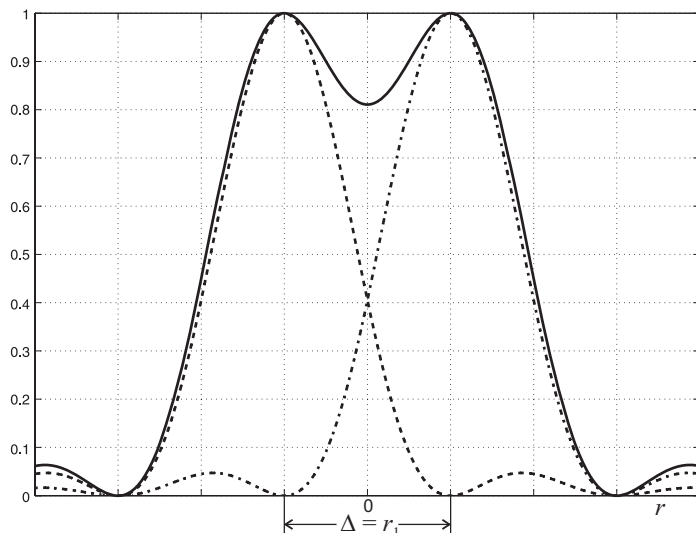


Рис. 3.16. Распределение интенсивностей от двух источников, когда  $\Delta = r_1$ :

$$\begin{aligned}
 \text{-----} & \quad I_1(r) = \text{sinc}^2(r + \Delta/2) \quad \text{— интенсивность 1-го источника,} \\
 \text{- \cdot \cdot \cdot} & \quad I_2(r) = \text{sinc}^2(r - \Delta/2) \quad \text{— интенсивность 2-го источника,} \\
 \text{—} & \quad I(r) = I_1(r) + I_2(r) \quad \text{— суммарная интенсивность}
 \end{aligned}$$

воспринимаемого человеческим глазом (глазом астронома). При этом учтено, что  $1 \text{ рад} = (180/\pi) \cdot 3600''$ .

Например, если  $D = 1 \text{ м} = 10^3 \text{ мм}$ , то  $\delta = 0.14 \text{ угл. сек}$  ( $\delta = 0.14''$ ).

Однако, во-первых, критерий Рэля является весьма субъективным (наблюдатель с острым зрением может различить источники при  $\Delta < r_1$ , а наблюдатель со слабым зрением различит источники лишь при  $\Delta > r_1$ , см. рис. 3.16). А во-вторых, этот критерий сформулирован для случая отсутствия погрешностей (шумов). При наличии же шумов (но без математической обработки) разрешающая способность инструмента (телескопа, спектроскопа, глаза) может заметно отличаться от предела, даваемого критерием Рэля. И, наконец, в-третьих, если привлечь достаточно эффективную методику для математической обработки результатов измерений с целью повышения разрешающей способности инструмента, то можно даже при наличии шумов заметно превзойти предел разрешения, даваемый критерием Рэля. Примеры этого — на рис. 3.9, 3.10, 3.14 и в публикациях [14, 26, 27, 33, 36, 39, 59, 61, 63], [102, с. 198, 199], [120, 135] и др., ср. также рис. 2.59g1 и рис. 2.61w\_alpha2, 2.35(2) и 2.38(4) и т. д.

О роли математики в повышении качества измерений великолепно высказался сам Рэлей, который в 1871 г. сформулировал следующую задачу (в связи с задачами спектроскопии): обработать математически некоторую измеренную прибором функцию (выходной сигнал) так, чтобы восстановить истинный сигнал, поступивший на вход прибора (входной сигнал), учитывая его аппаратную функцию (АФ). Сейчас эта задача называется *редукционной проблемой Рэля* [17, 33] и она заключается в редукции (приведении) измерений к идеальному измерительному устройству (устройству с бесконечно высокой разрешающей способностью).

Во времена Рэля казалось, что эта идея вполне осуществима и тогда можно будет чисто математическим путем устранять технические несовершенства различных измерительных устройств. Однако затем выяснилось (в начале XX века, во времена Адамара), что очень многие прикладные задачи в математическом отношении являются некорректными — чрезвычайно чувствительными даже к очень малым погрешностям измерений, ошибке оператора и т. д. В результате редукционная проблема Рэля математически оказалась значительно сложнее, чем представлялась раньше, и лишь с середины XX века она стала решаться достаточно эффективно.

Сформулируем математически редукционную проблему Рэля. Пусть  $f$  — результат измерений, или выходная функция (эхо-сигнал в функции времени в ЯМР-томографии, распределение интенсивности по искаженному изображению в функции линейных координат в иконике, спектр — распределение энергии по частоте  $\nu$  в спектроскопии и т. д.),  $y$  — входная (искомая, неизвестная) функция и пусть  $y$  и  $f$  связаны математически (а также физически и технически) соотношением:

$$Ay = f, \quad (3.29)$$

где  $A$  — некоторый математический оператор (интегральный, алгебраический, дифференциальный и т. д.), соответствующий физически и технически АФ, СЧ, ХН и т. д. Тогда решение уравнения (3.29) можно рассматривать как *обратную задачу* относительно  $y$ , т. е. как *задачу редукции*. Ее решение эффективными методами (регуляризации Тихонова, фильтрации Винера и т. д.) позволит повысить разрешающую способность измерительного устройства (разрешить близкие детали в изображении, разделить близкие линии в спектре и т. д.). Для повышения эффективности решения данной задачи целесообразно состыковать измерительное устройство с вычислительным устройством (компьютером, спецпроцессором и т. д.), в которое должно быть заложено программное обеспечение, реализующее тот или иной эффективный математический метод и численный алгоритм решения уравнения типа (3.29) [54, 55, 65, 124].

### Контрольные задания и вопросы

1. Что такое дискретный спектр? Объясните рис. 3.11 и 3.12 (схемы дискретного спектра), а также соотношения (3.19) и (3.20).

2. Почему (3.20) есть система линейно-нелинейных уравнений (СЛНУ)? Можно ли СЛНУ (3.20) считать системой нелинейных уравнений (СНУ)? Перечислите методы решения СНУ. Почему они неэффективны для решения СЛНУ? Перечислите методы решения СЛНУ.

3. Изложите кратко алгоритм интегральной аппроксимации решения СЛНУ.

4. Поясните численный пример, отображенный на рис. 3.13. Объясните основные операторы в первой и во второй части головной программы `AlgIntApp.m`. Поясните результат решения, приведенный на рис. 3.14.

5. Изложите некоторые полезные приемы, которые могут повысить скорость выполнения головных программ и  $m$ -функций в системе MatLab, и сравните MatLab с такими языками программирования, как Fortran и C/C++.

6. Сформулируйте критерий Рэлея. Какова его формулировка без использования математической обработки и что дает математическая обработка измерений?
7. Сформулируйте редуцированную проблему Рэлея. Почему ее стали эффективно реализовывать лишь со второй половины XX века?
8. Выведите формулу (3.28) из формулы (3.27), положив  $\lambda = 560$  нм.

# Литература

- [1] *Апарцин А.С.* Неклассические уравнения Вольтерра I рода: теория и численные методы. — Новосибирск: Наука, 1999. 193 с.
- [2] *Арефьева М.В., Сысоев А.Ф.* Быстрые регуляризирующие алгоритмы цифрового восстановления изображений // Вычислит. методы и программирование. 1983. Вып. 39. С. 40–55.
- [3] *Арсеньев Д.Г., Иванов В.М., Корневский М.Л.* Адаптивное управление в стохастических методах вычислительной математики и механики. — СПб.: Наука, 2006. 416 с.
- [4] *Бакушинский А.Б., Гончарский А.В.* Некорректные задачи. Численные методы и приложения. — М.: Изд-во МГУ, 1989. 199 с.
- [5] *Бамбер Дж., Тристам М.* Ультразвуковая диагностика // Физика визуализации изображений в медицине / Под ред. С. Уэбба. — М.: Мир, 1991. Т. 2. С. 5–104.
- [6] *Бейтс Р., Мак-Доннелл М.* Восстановление и реконструкция изображений. — М.: Мир, 1989. 336 с.
- [7] *Белов И.А., Сизиков В.С.* Программа реконструкции смазанных и дефокусированных электронных изображений. — Свидетельство об официальной регистрации программы для ЭВМ № 2001610743. Зарегистрир. 18.06.2001.
- [8] *Бельфер И.К., Непомнящих И.А.* Сейсмическая томография. Разведочная геофизика: обзор. — М.: Изд-во ВИЭМС, 1988. 70 с.
- [9] *Березин И.С., Жидков Н.П.* Методы вычислений. В 2-х т. Изд-е 3-е. Т. 1. — М.: Наука, 1966. 632 с.
- [10] *Бойков И.В.* Оптимальные методы приближенного вычисления интегралов и приближенного решения интегральных уравнений. Уч. пособие. — Пенза: Изд-во ППИ, 1981. 106 с.
- [11] *Бойков И.В.* Приближенное решение уравнений в свертках и их применение. Уч. пособие. — Пенза: Изд-во ППИ, 1985. 65 с.



- [12] *Боттомли П.А.* ЯМР-интроскопия: методы и применение (обзор) // Приборы для научных исследований. 1982. № 9. С. 3–28.
- [13] *Бочек И.А.* О вычислении коэффициентов Ритца и Галеркина методом Монте-Карло // Ж. вычисл. матем. и матем. физики. 1967. Т. 7. № 1. С. 172–177.
- [14] *Брагинская Т.Г., Клобин В.В.* Решение обратной задачи спектроскопии оптического смещения методом регуляризации Тихонова. Препринт № 855. — Л.: Изд-во ЛИЯФ, 1983. 60 с.
- [15] *Брейсуэлл Р.* Преобразование Хартли. — М.: Мир, 1990. 175 с.
- [16] *Бронштейн И.Н., Семендяев К.А.* Справочник по математике для инженеров и учащихся втузов. Изд-е 13-е. — М.: Наука, 1986. 544 с.
- [17] *Василенко Г.И.* Теория восстановления сигналов: О редукции к идеальному прибору в физике и технике. — М.: Сов. радио, 1979. 272 с.
- [18] *Василенко Г.И., Тараторин А.М.* Восстановление изображений. — М.: Радио и связь, 1986. 304 с.
- [19] *Васильев В.Н., Гуров И.П.* Компьютерная обработка сигналов в приложении к интерферометрическим системам.— СПб.: БХВ – Санкт-Петербург, 1998. 240 с.
- [20] *Васин В.В.* Аппроксимация негладких решений линейных некорректных задач // Тр. Ин-та матем. и мех. УрО РАН. 2006. Т. 12. № 1. С. 64–77.
- [21] *Васин В.В., Агеев А.Л.* Некорректные задачи с априорной информацией. — Екатеринбург: УИФ "Наука", 1993. 262 с.
- [22] *Верлань А.Ф., Сизиков В.С.* Интегральные уравнения: методы, алгоритмы, программы. — Киев: Наук. думка, 1986. 544 с.
- [23] *Верлань А.Ф., Сизиков В.С., Мосенцова Л.В.* Метод вычислительных экспериментов для решения интегральных уравнений в обратной задаче спектроскопии // Электрон. моделирование. 2011. Т. 33. № 2. С. 3–12.
- [24] *Воронцов Е.А.* Методы, алгоритмы и технические средства оперативной оценки качества изображений. Автореф. дис. ... канд. техн. наук. — СПб.: СПбГУ ИТМО, 2006. 12 с.
- [25] *Воронцов Е.А., Мусалимов В.М.* Проблемы оценивания динамической остроты зрения // Изв. вузов. Приборостроение. 2010. Т. 53. № 2. С. 57–61.
- [26] *Воскобойников Ю.Е.* Комбинированный нелинейный алгоритм восстановления контрастных изображений при неточно заданной аппаратной функции // Автометрия. 2007. Т. 43. № 6. С. 3–18.

- [27] *Воскобойников Ю.Е., Литасов В.А.* Устойчивый алгоритм восстановления изображения при неточно заданной аппаратной функции // *Автометрия*. 2006. Т. 42. № 6. С. 3–15.
- [28] *Воскобойников Ю.Е., Мухина И.Н.* Локальный регуляризирующий алгоритм восстановления контрастных сигналов и изображений // *Автометрия*. 2000. № 3. С. 45–53.
- [29] *Воскобойников Ю.Е., Преображенский Н.Г., Седелников А.И.* Математическая обработка эксперимента в молекулярной газодинамике. — Новосибирск: Наука, 1984. 240 с.
- [30] *Галайдин П.А., Замятин А.И., Иванов В.А.* Основы магниторезонансной томографии. Уч. пособие. — СПб.: Изд-во ИТМО, 1998. 24 с.
- [31] *Галайдин П.А., Иванов В.А., Марусина М.Я.* Расчет и проектирование электромагнитных систем магниторезонансных томографов. Уч. пособие. — СПб.: Изд-во СПбГУ ИТМО, 2004. 87 с.
- [32] *Гахов Ф.Д., Черский Ю.И.* Уравнения типа свертки. — М.: Наука, 1978. 296 с.
- [33] *Глазов М.В., Болохова Т.А.* Решение редуцированной проблемы Рэлея с использованием различных модификаций метода регуляризации // *Оптика и спектроскопия*. 1989. Т. 67. Вып. 3. С. 533–537.
- [34] *Голов И.Н., Иванов В.А., Сизиков В.С.* Интерпретация обратной задачи гравиметрии как задачи гравитационной томографии и внутривидения Земли // *Научное приборостроение*. 2003. Т. 13. № 2. С. 27–33.
- [35] *Гонсалес Р., Вудс Р.* Цифровая обработка изображений. — М.: Техносфера, 2006. 1072 с.
- [36] *Гонсалес Р., Вудс Р., Эддинс С.* Цифровая обработка изображений в среде MATLAB. — М.: Техносфера, 2006. 616 с.
- [37] *Гончарский А.В., Черепащук А.М., Ягола А.Г.* Численные методы решения обратных задач астрофизики. — М.: Наука, 1978. 336 с.
- [38] *Гончарский А.В., Черепащук А.М., Ягола А.Г.* Некорректные задачи астрофизики. — М.: Наука, 1985. 352 с.
- [39] *Горшков А.В.* Улучшение разрешения изображений при обработке данных физического эксперимента и нахождение неизвестной аппаратной функции по программам пакета REIMAGE // *Приборы и техника эксперимента*. 1995. № 2. С. 68–78.
- [40] *Грузман И.С., Киричук В.С., Косых В.П., Перетягин Г.И., Спектор А.А.* Цифровая обработка изображений в информационных системах. Уч. пособие. — Новосибирск: Изд-во НГТУ, 2002. 352 с.

- [41] *Дайнеко М.В.* Реконструкция смазанных и зашумленных изображений методами регуляризации и усечения в технических системах обработки информации. Автореф. дис. ... канд. техн. наук. — Санкт-Петербург, 2011. 23 с.
- [42] *Дайнеко М.В., Сизиков В.С.* Восстановление смазанных под углом и зашумленных изображений без учета граничных условий // Научно-техн. Вестник СПбГУ ИТМО. 2010. Вып. 4 (68). С. 28–32.
- [43] *Дайнеко М.В., Сизиков В.С.* Реконструкция смазанных под углом полутонновых изображений методом регуляризации. — Свидетельство о гос. регистрации программы для ЭВМ № 2011613219. Зарегистрир. 25.04.2011.
- [44] *Дьяконов В.* MATLAB 6: учебный курс. — СПб.: Питер, 2001. 592 с.
- [45] *Дьяконов В., Абраменкова И.* MATLAB. Обработка сигналов и изображений. Специальный справочник. — СПб.: Питер, 2002. 608 с.
- [46] *Журавель И.М.* Краткий курс теории обработки изображений, 2008, <http://matlab.exponenta.ru/imageprocess/book2/index.php>
- [47] *Иванов В.А.* Способ определения внутреннего строения материальных объектов. А. с. № 1112266 // Открытия, изобретения. 1984. № 33 (Приоритет от 21.03.60).
- [48] *Иванов В.А.* Внутривидение (ЯМР-томография). — Л.: Знание, 1989. 28 с.
- [49] *Иванов В.А., Марусина М.Я., Руценко Н.Г., Сизиков В.С.* Реконструкция МР-изображений с учетом неоднородностей полей // Научное приборостроение. 2003. Т. 13. № 2. С. 17–21.
- [50] *Иванов В.К., Васин В.В., Танана В.П.* Теория линейных некорректных задач и ее приложения. — М.: Наука, 1978. 206 с.
- [51] *Ивнинг М.* Adobe Photoshop CS4 для фотографов. — М.: Рус. Редакция, СПб.: БХВ – Петербург, 2009. 704 с.
- [52] *Иконика.* Сборник статей. — М.: Наука, 1968, 1970, 1975.
- [53] *Кей С.М., Марпл С.Л.* Современные методы спектрального анализа (обзор) // Труды Ин-та инж. по электротехнике и радиоэлектрон. 1981. Т. 69. № 11. С. 5–51.
- [54] *Кирьянов К.А.* Инструментальная реализация алгоритмов реконструкции искаженных изображений // Труды 20-й Междунар. конф. "GraphiCon-2010". — СПб.: Изд-во СПбГУ ИТМО, 2010. С. 188–191.
- [55] *Кирьянов К.А., Сизиков В.С.* Применение сигнальных микропроцессоров в задачах реконструкции искаженных изображений // Изв. вузов. Приборостроение. 2011. Т. 54. № 7. С. 20–26.

- [56] *Коренблюм Б.И., Тетельбаум С.И., Тютин А.А.* Об одной схеме томографии // Изв. вузов. Радиофизика. 1958. Т. 1. № 3. С. 151–157.
- [57] *Корн Г., Корн Т.* Справочник по математике для научных работников и инженеров. — М.: Наука, 1968. 720 с. С. 1270–1284.
- [58] *Котельников В.А.* Теория потенциальной помехоустойчивости. — М.–Л.: Госэнергоиздат, 1956. 151 с.
- [59] *Кочиков И.В., Курамшина Г.М., Пентин Ю.А., Ягола А.Г.* Обратные задачи колебательной спектроскопии. — М.: Изд-во МГУ, 1993. 204 с.
- [60] *Красильников Н.Н.* Реставрация изображений с учетом их структуры // Оптический журнал. 2009. Т. 76. № 2. С. 7–12.
- [61] *Крауляня Э.К., Лица С.Я., Пикалов В.В., Скудра А.Я.* К проблеме исследования атомной сенсibilизированной флуоресценции по контурам спектральных линий // Некорректные обратные задачи атомной физики / Под ред. Н.Г. Преображенского. — Новосибирск: Изд-во ИТПМ, 1976. С. 61–72.
- [62] *Кривых А.В.* О решении обратной задачи спектроскопии методом регуляризации // Сб. трудов конф. молодых ученых. Вып. 2. Биомедиц. технологии, мехатроника и робототехника / Гл. ред. В.Л. Ткалич. — СПб.: СПбГУ ИТМО, 2009. С. 41–45.
- [63] *Кривых А.В., Сизиков В.С.* О решении обратной задачи спектроскопии для непрерывных спектров методом регуляризации // Материалы международ. конф. "XXXVIII Неделя науки СПбГПУ". Ч. XIII (ИМОП) / Под ред. А.И. Рудского. — СПб.: СПбГПУ, 2009. С. 177–179.
- [64] *Кривых А.В., Сизиков В.С.* Обработка дискретных спектров с помощью алгоритма интегральной аппроксимации // Научно-техн. Вестник СПбГУ ИТМО. 2011. Вып. 5 (75). С. 14–18.
- [65] *Кухарев Г.А., Тропченко А.Ю., Шмерко В.П.* Систематические процессоры для обработки сигналов. — Минск: Беларусь, 1988. 127 с.
- [66] *Лаврентьев М.М., Романов В.Г., Шишатский С.П.* Некорректные задачи математической физики и анализа. — М.: Наука, 1980. 288 с.
- [67] *Ландсберг Г.С.* Оптика. Уч. пособие для вузов. Изд-е 6-е. — М.: Физматлит, 2006. 848 с.
- [68] *Лич М.* Получение ЯМР-изображений с пространственной локализацией // Физика визуализации изображений в медицине / Под ред. С. Уэбба. — М.: Мир, 1991. Т. 2. С. 105–231.

- [69] *Марусина М.Я., Казначеева А.О.* Современные виды томографии. Уч. пособие. — СПб.: Изд-во СПбГУ ИТМО, 2006. 152 с.
- [70] *Марусина М.Я., Казначеева А.О.* Современное состояние и перспективы развития томографии // Научно-техн. Вестник СПбГУ ИТМО. 2007. Вып. 42. С. 3–13.
- [71] *Математическая Энциклопедия.* В 5-ти т. / Гл. ред. И.М. Виноградов. — М.: Сов. Энциклопедия. 1977.
- [72] *Мельников Г.И.* Динамика нелинейных механических и электромеханических систем. — Л.: Машиностроение, 1975. 200 с.
- [73] *Методы компьютерной обработки изображений* / Под ред. В.А. Сойфера. — М.: Физматлит, 2001. 784 с.
- [74] *Мирджамолов Р.К.* Восстановление цветных смазанных и зашумленных изображений методом регуляризации с использованием приема усечения // Сб. трудов конф. молодых ученых. Вып. 2. Биомедиц. технологии, мехатроника и робототехника / Гл. ред. В.Л. Ткалич. — СПб.: СПбГУ ИТМО, 2009. С. 28–34.
- [75] *Монтгомери Д.Б.* Получение сильных магнитных полей с помощью соленоидов. — М.: Мир, 1971. 360 с.
- [76] *Морозов В.А.* Регулярные методы решения некорректно поставленных задач. — М.: Наука, 1987. 240 с.
- [77] *Мосенцова Л.В.* Реализация метода моделирования (вычислительных экспериментов) для решения уравнений Фредгольма I рода в системе МАТЛАВ // Сб. тезисов конф. "Интегральные уравнения – 2009". — Киев: Изд-во ИПМЭ, 2009. С. 110–112.
- [78] *Наттерер Ф.* Математические аспекты компьютерной томографии.— М.: Мир, 1990. 288 с.
- [79] *Неронов Ю.И.* Магнитный резонанс в томографии и в спектральных исследованиях тканей живого организма. Уч. пособие. — СПб.: Изд-во СПбГУ ИТМО, 2007. 120 с.
- [80] *Неронов Ю.И., Сизиков В.С., Соколов Д.Ю.* Сравнение трех методов синтеза однородного поля перманентного магнита в МР-томографии // Научно-техн. Вестник СПбГУ ИТМО. 2007. Вып. 45. С. 190–202.
- [81] *Обработка изображений и цифровая фильтрация* / Под ред. Т. Хуанга. — М.: Мир, 1979. 320 с.
- [82] *Остриков В.Н., Плахотников О.В.* Идентификация функции рассеяния точки канала наблюдения по калибрующему изображению посредством метода наименьших квадратов // Оптич. журнал. 2006. Т. 73. № 2. С. 26–30.

- [83] *Перчик Е.* Методология синтеза знаний: преодоление фактора некорректности задач математического моделирования. — Харьков, 2004, 205 с. <http://www.pelbook.narod.ru>
- [84] *Пиблз, Берковиц.* Многоцелевой моноимпульсный радиолокатор // Зарубеж. радиоэлектрон. 1969. № 10. С. 3–17.
- [85] *Пикалов В.В., Мельникова Т.С.* Томография плазмы (Низкотемпературная плазма, т. 13). — Новосибирск: Наука. Сиб. изд. фирма РАН, 1995. 229 с.
- [86] *Пикалов В.В., Непомнящий А.В.* Итерационный алгоритм с вэйвлет-фильтрацией в задаче двумерной томографии // Вычислит. методы и программирование. 2003. Т. 4. С. 244–253.
- [87] *Пикалов В.В., Преображенский Н.Г.* Реконструктивная томография в газодинамике и физике плазмы. — Новосибирск: Наука, 1987. 239 с.
- [88] *Преображенский Н.Г., Пикалов В.В.* Неустойчивые задачи диагностики плазмы. — Новосибирск: Наука, 1982. 238 с.
- [89] *Преображенский Н.Г., Седелников А.И.* Оптимизация измерений, проводимых с помощью реальных спектральных приборов. Препринт № 15. — Новосибирск: Изд-во ИТПМ, 1980. 23 с.
- [90] *Прохоров А.А., Прокди Р.Г., Рупп А.П., Финков М.В. и др.* Самоучитель Photoshop CS4. Официальная русская версия. Книга + видеокурс. — СПб.: Наука и Техника, 2009. 384 с.
- [91] *Прэтт У.* Цифровая обработка изображений. В 2-х т. — М.: Мир, 1982. 790 с.
- [92] *Пытьев Ю.П., Чуличков А.И.* Прибор + ЭВМ = новые возможности. — М.: Знание, 1983. 64 с.
- [93] *Радченко С.Г.* Устойчивые методы оценивания статистических моделей. — Киев: ИП "Санспарель", 2005. 504 с. Знание, 1983.
- [94] *Раутиан С.Г.* Реальные спектральные приборы // Успехи физических наук. 1958. Т. 66. Вып. 3. С. 475–517.
- [95] *Римских М.В.* Использование приема усечения для реконструкции смазанных под углом и зашумленных изображений // Сб. трудов конф. молодых ученых. Вып. 2. Биомедиц. технологии, мехатроника и робототехника / Гл. ред. В.Л. Ткалич. — СПб.: СПбГУ ИТМО, 2009. С. 21–27.
- [96] *Римских М.В., Евсеев В.О., Сизиков В.С.* Реконструкция смазанных изображений различными методами // Оптич. журнал. 2007. Т. 74. № 11. С. 53–57.
- [97] *Ринкк П.А.* Магнитный резонанс в медицине. Основной учебник Европейского форума по магнитному резонансу. — М.: ГЭОТАР-МЕД, 2003. 256 с.

- [98] *Свиридова К.* Photoshop CS4 – это просто. Экспресс-методы обработки фотографий. — СПб.: БХВ – Петербург, 2010. 384 с.
- [99] *Сизиков В. С.* О моделировании некоторых некорректных задач с использованием принципов подобия // Электрон. моделирование. 1981. Вып. 6. С. 3–8.
- [100] *Сизиков В. С.* Обобщенный метод редукции измерений. I, III // Электрон. моделирование. 1991. Т. 13. № 4. С. 7–14. № 6. С. 3–9.
- [101] *Сизиков В. С.* Использование регуляризации для устойчивого вычисления преобразования Фурье // Ж. вычисл. матем. и матем. физики. 1998. Т. 38. № 3. С. 376–386.
- [102] *Сизиков В. С.* Математические методы обработки результатов измерений: Учебник для вузов. — СПб.: Политехника, 2001. 240 с. (Эл. вариант: *Сизиков В. С.* Устойчивые методы обработки результатов измерений. Эл. учебник, 1999. — <http://dsp-book.narod.ru/SIZIKOV.pdf>).
- [103] *Сизиков В. С.* О способах невязки при решении некорректных задач // Ж. вычисл. матем. и матем. физики. 2003. Т. 43. № 9. С. 1294–1312.
- [104] *Сизиков В. С.* О решении некорректных, неклассических, нестандартных и сингулярных интегральных уравнений // Сб. тезисов конф. "Интегральные уравнения – 2009". — Киев: Изд-во ИПМЭ, 2009. С. 17–21.
- [105] *Сизиков В. С.* Применение аппарата интегральных уравнений для реконструкции искаженных изображений // Сб. тезисов конф. "Интегральные уравнения – 2009". — Киев: Изд-во ИПМЭ, 2009. С. 128–130.
- [106] *Сизиков В. С.* Интегральные уравнения и новый прием "усечение–размытие–поворот" в реконструкции искаженных изображений // Proc. Intern. Conf. "Integral Equations – 2010". — Lviv: PAIS, 2010. P. 138–142.
- [107] *Сизиков В. С.* Реконструкция смазанных и зашумленных изображений (RSNI). — Свидетельство о гос. регистрации программы для ЭВМ № 2010613829. Зарегистрир. 10.06.2010.
- [108] *Сизиков В. С.* Прием "усечение–размытие–поворот" для восстановления искаженных изображений // Оптич. журнал. 2011. Т. 78. № 5. С. 18–26.
- [109] *Сизиков В. С., Белов И. А.* Реконструкция смазанных и дефокусированных изображений методом регуляризации // Оптич. журнал. 2000. Т. 67. № 4. С. 60–63.
- [110] *Сизиков В. С., Кривых А. В.* Применение способа эталонных примеров при решении обратной задачи спектроскопии методом регуляризации // Изв. вузов. Приборостроение. 2011. Т. 54. № 9. С. 44–51.

- [111] *Сизиков В.С., Кузьмин А.В., Козаченко А.В.* Обработка дефокусированных изображений методами двухмерного преобразования Хартли и регуляризации Тихонова // Изв. вузов. Приборостроение. 1999. Т. 42. № 8. С. 12–16.
- [112] *Сизиков В.С., Римский М.В., Мирджамолов Р.К.* Реконструкция смазанных и зашумленных изображений без использования граничных условий // Оптический журнал. 2009. Т. 76. № 5. С. 38–46.
- [113] *Сизиков В.С., Российская М.В., Козаченко А.В.* Обработка смазанного изображения методами дифференцирования, преобразования Хартли и регуляризации Тихонова // Изв. вузов. Приборостроение. 1999. Т. 42. № 7. С. 11–15.
- [114] *Старков В.Н.* Конструктивные методы вычислительной физики в задачах интерпретации. — Киев: Наук. думка, 2002. 264 с.
- [115] *Старостенко В.И.* Устойчивые численные методы в задачах гравиметрии. — Киев: Наук. думка, 1978. 228 с.
- [116] *Суинделл Б., Уэбб С.* Рентгеновская трансмиссионная томография // Физика визуализации изображений в медицине / Под ред. С. Уэбба. — М.: Мир, 1991. Т. 1. С. 138–173.
- [117] *Тихонов А.Н., Арсенин В.Я.* Методы решения некорректных задач. Изд-е 3-е. — М.: Наука, 1986. 288 с.
- [118] *Тихонов А.Н., Арсенин В.Я., Рубашов И.Б., Тимонов А.А.* Первый советский компьютерный томограф // Природа. 1984. № 4. С. 11–21.
- [119] *Тихонов А.Н., Арсенин В.Я., Тимонов А.А.* Математические задачи компьютерной томографии. — М.: Наука, 1987. 160 с.
- [120] *Тихонов А.Н., Гончарский А.В., Степанов В.В.* Обратные задачи обработки фотоизображений // Некорректные задачи естествознания / Под ред. А.Н. Тихонова, А.В. Гончарского. — М.: Изд-во МГУ, 1987. С. 185–195.
- [121] *Тихонов А.Н., Гончарский А.В., Степанов В.В., Ягола А.Г.* Численные методы решения некорректных задач. — М.: Наука, 1990. 232 с.
- [122] *Тозик В.Т., Меженнин А.В., Кротова А.Ю., Зеленковский А.Л.* Использование 3D виртуальных сред для верификации методов распознавания и реконструкции // Труды XIII Всерос. научно-методич. конф. "Телематика'2008". — СПб, 2008.
- [123] *Троцкий И.Н.* Статистическая теория томографии. — М.: Радио и связь, 1989. 240 с.
- [124] *Тропченко А.Ю.* Аппаратные средства для цифровой обработки сигналов. Уч.-методич. пособие по дисциплине "Методы обработки сигналов и изображений". — СПб.: Изд-во СПбГУ ИТМО, 2005. 138 с.



- [125] *Турчин В. Ф.* Решение уравнения Фредгольма I рода в статистическом ансамбле гладких функций // *Ж. вычисл. матем. и матем. физики*. 1967. Т. 7. № 6. С. 1270–1284.
- [126] *Тюрин А. М.* Введение в теорию статистических методов в гидроакустике. — Л.: Изд-во ВМОЛА, 1963. 252 с.
- [127] *Фалькович С. Е., Коновалов Л. Н.* Разрешение неизвестного числа сигналов // *Радиотехника и электрон.* 1982. Т. 27. № 1. С. 92–97.
- [128] *Физический* энциклопедический словарь / Гл. ред. А. М. Прохоров. — М.: Сов. Энциклопедия, 1984. 944 с.
- [129] *Фисенко В. Т., Фисенко Т. Ю.* Компьютерная обработка и распознавание изображений. Уч. пособие. — СПб.: Изд-во СПбГУ ИТМО, 2008. 192 с.
- [130] *Фридман В. М.* Метод последовательных приближений для интегрального уравнения Фредгольма 1-го рода // *Успехи математических наук*. 1956. Т. 11. Вып. 1(67). С. 233–234.
- [131] *Хермен Г.* Восстановление изображений по проекциям. Основы реконструктивной томографии. — М.: Мир, 1983. 352 с.
- [132] *Химмельблау Д.* Прикладное нелинейное программирование. — М.: Мир, 1975. 536 с.
- [133] *Хорт В.* "Четыре третьих" в цифровой фотографии: значение имеет не только размер // *Наука и жизнь*. 2009. № 2. С. 114–116.
- [134] *Хорт В.* От камеры-обскуры – к телецентрическому объективу // *Наука и жизнь*. 2009. № 3. С. 98–101.
- [135] *Чейссон Э. Дж.* Первые результаты с космического телескопа "Хаббл" // *В мире науки*. 1992. № 8. С. 6–14.
- [136] *Шлихт Г. Ю.* Цифровая обработка цветных изображений. — М.: ЭКОМ, 1997. 336 с.
- [137] *Эндрюс Г.* Применение вычислительных машин для обработки изображений. — М.: Энергия, 1977. 160 с.
- [138] *Эндрюс Ф.* Цифровая фотография. Практическое руководство. — М.: ЗАО "РОСМЭН-ИЗДАТ", 2005. 192 с.
- [139] *Эрнст Р., Боденхаузен Дж., Вокаун А.* ЯМР в одном и двух измерениях. — М.: Мир, 1990.
- [140] *Ягола А. Г., Кошев Н. А.* Восстановление смазанных и дефокусированных цветных изображений // *Вычислит. методы и программирование*. 2008. Т. 9. С. 207–212.

- [141] *Ядерный* магнитный резонанс. Уч. пособие / Под ред. П.М. Бородина. — Л.: Изд-во ЛГУ, 1982. 344 с.
- [142] *Яне Б.* Цифровая обработка изображений. — М.: Техносфера, 2007. 584 с.
- [143] *Ярославский Л.П.* Введение в цифровую обработку изображений. — М.: Сов. радио, 1979. 312 с.
- [144] *Arico A., Donatelli M., Nagy J., Serra-Capizzano S.* The anti-reflective transform and regularization by filtering. Technical report TR-2007-006-A, <ftp://ftp.mathcs.emory.edu/pub/techreport/TR-2007-006-A.pdf>
- [145] *Aubert G., Kornprobst P.* Mathematical Problems in Image Processing. Partial Differential Equations and the Calculus of Variations. 2nd ed. Vol. 147 of Applied Mathematical Sciences. — New York: Springer, 2006. 377 pp.
- [146] *Bracewell R.N., Riddle A.C.* Inversion of fan-beam scans in radio astronomy // *Astrophys. J.* 1967. Vol. 150. P. 427–434.
- [147] *Cho Z.H., Jones J.P., Singh M.* Foundations of medical imaging. — New York: Wiley, 1993. 586 pp.
- [148] *Christiansen M., Hanke M.* Deblurring methods using antireflective boundary conditions, 2006, <http://www.numerik.mathematik.uni-mainz.de/\,hanke/christiansen.pdf>
- [149] *Donatelli M.* Fast transforms for high order boundary conditions, 2009, <http://arxiv.org/abs/0906.2704v1.pdf>
- [150] *Donatelli M., Estatico C., Martinelli A., Serra-Capizzano S.* Improved image deblurring with anti-reflective boundary conditions and re-blurring // *Inverse problems.* 2006. Vol. 22. P. 2035–2053.
- [151] *Engl H., Hanke M., Neubauer A.* Regularization of Inverse Problems. — Dordrecht: Kluwer, 1996. 328 pp.
- [152] *Farsiu S., Robinson M.D., Elad M., Milanfar P.* Fast and robust multiframe super resolution // *IEEE Transactions on Image Processing.* 2004. Vol. 13. No. 10. P. 1327–1344.
- [153] *Flechl T., Jäger H., Obernberger I.* Experimental verification of gas spectra calculated for high temperatures using the HITRAN/HITEMP database // *J. Physics D: Applied Physics.* 2002. Vol. 35. P. 3138–3144.
- [154] *Golub G.H., Pereyra V.* The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate // *SIAM J. Numer. Anal.* 1973. Vol. 10. P. 413–432.

- 
- [155] *Hegland M.* Error bounds for spectral enhancement which are based on variable Hilbert scale inequalities // *J. Integral Equations Appl.* 2010. Vol. 22. No. 2. P. 285–312.
- [156] *Hounsfield G.N.* Computerised transverse axial scanning (tomography). Part 1: Description of system // *Br. J. Radiol.* 1973. Vol. 46. P. 1016–1022.
- [157] *Introduction to MR Tomography and in-vivo Spectroscopy. Basic Principles* // Abstract of BRUKER Medical Report 83/1 and 85/1, 1986. 15 pp.
- [158] *Kojdecki M.A.* New criterion of regularisation parameter choice in Tikhonov's method // *Biuletyn WAT (Biul. Mil. Univ. Technol.)*. 2000. Vol. XLIX. No. 1(569). P. 47–126.
- [159] *Kumar A., Welte D., Ernst R.R.* NMR Fourier zeugmatography // *J. Magnetic Resonance*. 1975. Vol. 18. P. 69–83.
- [160] *Landweber L.* An iteration formula for Fredholm integral equations of the first kind // *Amer. J. Math.* 1951. Vol. 73. No. 3. P. 615–624.
- [161] *Lauterbur P.C.* Image formation by induced local interactions: examples of employing nuclear magnetic resonance // *Nature*. 1973. Vol. 242. P. 190–191.
- [162] *Mansfield P.* Multi-planar image formation using NMR spin echoes // *J. Physics C: Solid State Physics*. 1977. Vol. 10. P. 55–58.
- [163] *Mullen K.M., van Stokkum I.H.M.* The variable projection algorithm in time-resolved spectroscopy, microscopy and mass spectrometry applications // *Numerical Algorithms*. 2009. Vol. 51. No. 3. P. 319–340.
- [164] *Nagy J.G., Palmer K.M., Perrone L.* Iterative methods for image deblurring: A Matlab object oriented approach // *Numer. Algorithms*. 2004. Vol. 36. P. 73–93.
- [165] *Nasonov A.V., Krylov A.S.* Adaptive image deringing // *Proceedings of the Conf. GraphiCon'2009, Moscow, 2009*. P. 151–158.
- [166] *Ng M.K., Chan R.H., Tang W.-C.* A fast algorithm for deblurring models with Neumann boundary conditions // *SIAM J. Sci. Comput.* 1999. Vol. 21. P. 851–866.
- [167] *Palmer K., Nagy J., Perrone L.* Iterative methods for image restoration: a Matlab object oriented approach, 2002,  
<ftp://ftp.mathcs.emory.edu/pub/nagy/papers/RestoreTools1.pdf>
- [168] *Petrov Yu.P., Sizikov V.S.* Well-Posed, Ill-Posed, and Intermediate Problems with Applications. — Leiden–Boston: VSP, 2005. 234 p.
- [169] *Radon J.* Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten // *Ber. Verh. Sächs. Akad. Wiss. Leipzig Math. Phys. Kl.* 1917. B. 69. S. 262–277.

- [170] *Ramachandran G.N., Lakshminarayanan A.V.* Three-dimensional reconstruction from radiographs and electron micrographs: application of convolutions instead of Fourier transforms // Proc. Nat. Acad. Sci. USA. 1971. Vol. 68. No. 9. P. 2236–2240.
- [171] *Rudin L., Osher S., Fatemi E.* Nonlinear total variation based noise removal algorithms // Phys. D. 1992. Vol. 60. P. 259–268.
- [172] *Russ J.C.* The Image Processing Handbook. 5th ed. — Boca Raton: CRC Press, 2007. 818 pp.
- [173] *Saisse M., Rousselet K., Lazarides E.* Modeling technique for the Hubble Space Telescope wave-front deformation // Applied Optics. 1995. Vol. 34. No. 13. P. 2278–2283.
- [174] *Serra-Capizzano S.* A note on antireflective boundary conditions and fast deblurring methods // SIAM J. Sci. Comput. 2003. Vol. 25. P. 1307–1325.
- [175] *Shchekotin D., Sizikov V.* Using regularization and a priori information to enhance tomographic image quality // Imaging Sci. Journal. 2006. Vol. 54, No. 3, P. 147–151.
- [176] *Shepp L.A., Logan B.F.* The Fourier reconstruction of a head section // IEEE Trans. Nucl. Sci. 1974. Vol. NS-21. P. 21–43.
- [177] *Tourin R.H., Krakow B.* Applicability of infrared emission and absorption spectra to determination of hot gas temperature profiles // Applied Optics. 1965. Vol. 4. No. 2. P. 237–242.
- [178] *Wendykier P., Nagy J.G.* Large-scale image deblurring in Java. Technical Report TR-2008-002, <http://www.mathcs.emory.edu/technical-reports/techrep-00127.pdf>
- [179] *Wendykier P., Nagy J.G.* Image Processing on Modern CPUs and GPUs. Technical Report TR-2008-023, <http://www.mathcs.emory.edu/technical-reports/techrep-00148.pdf>
- [180] *Wheeler F.W., Liu X., Tu P.H.* Multi-frame super-resolution for face recognition, 2007, <http://utopia.csis.pace.edu/cs615/btas2010/BTAS/BTAS%20'07/Papers/Paper67.pdf>

# Предметный указатель

- Аберрации оптических систем 72
- Адаптивная фильтрация Винера 117
- Алгоритм интегральной аппроксимации решения СЛНУ 220
- Аналогия с бегунами по дорожке 55
- Ансамбль протонов 43, 45
  - реализаций 93
- Апертура 74, 159, 228
- Аппаратная функция (АФ) 14, 16, 203, 211, 221
- Аппроксимация 220
  
- Белый шум 35, 87, 174
- Быстрое преобразование Фурье (БПФ) 87
- Быстрый алгоритм реконструкции смазанных изображений 137
  
- Вариация функции  $V_a^b(w)$ ,  $V(w)$  92
- Вектор ядерной намагниченности ансамбля протонов  $\mathbf{M}$  43, 45
- Величина смаза  $\Delta$  13
- Внеграницные условия (off-boundary conditions, OBCs) 81, 163
- Внешняя m-функция 81
  - — `deconvblind.m` 81
  - — `deconvlucy.m` 81
  - — `deconvreg.m` 81
  - — `deconvwnr.m` 81, 93, 104, 137, 176
  - — `edgetaper.m` 84, 113, 81, 168
  - — `fspecial.m` 104, 122, 163
  - — `imfilter.m` 104, 122, 163
  - — `imnoise.m` 96
  - — `imshow.m` 137
  - — `medfilt2.m` 118, 180
  - — `wiener2.m` 118, 180
  
- Восстановление (restoration) дефокусированных изображений 156
  - зашумленных изображений 94
  - изображений 69
  - искаженных изображений 68
  - смазанных изображений 74, 94
  - спектра в спектроскопии 205, 216
- Временной процесс 11
- Время поперечной релаксации  $T_2$ ,  $T_2^*$  46, 50
  - продольной релаксации  $T_1$  46, 49, 50, 50
- Встроенные (внутренние) функции 81
- Входной процесс (сигнал) 14, 16
- Выбор параметра регуляризации  $\alpha$  99, 209
- Вывод интегрального уравнения 76
- Выходной процесс 15, 16
- Вычислительное устройство 11
  
- Гауссиана 160, 204, 212
- Гиромагнитное отношение  $\gamma$  42, 49
- Головные программы в системе MatLab 81, 96, 98, 169, 170, 213, 221
- Градиентные поля  $G_x$ ,  $G_y$ ,  $G_z$  56, 57
- Граница  $B$  исходного изображения 82
- "Граничные условия" (boundary conditions, BCs) 80, 122, 162
- Графический редактор PhotoShop 195
  
- Датчик случайных чисел `randn` 96
- Движение магнитного момента протона в магнитном поле 44, 44
- Двойной поворот изображения 143
- Детектор (приемник) 22
- Дефокусирование 156
- Дефокусированное (out-of-focus, defocused) изображение 156, 171, 173

- Дискретизация 220  
 — изображения 68  
 Дискретный спектр 217  
 Дифракционные ограничения 227  
  
**Задача некорректная** (ill-posed, incorrect) 27, 78, 87, 91, 174, 206  
 — неустойчивая (unstable) 91, 206  
 — обратная (inverse) 15, 86, 103, 127, 131, 141, 172  
 — прямая (direct) 11, 102, 126, 130, 142, 162  
 — спектроскопии 199  
 — редукции к идеальному спектральному прибору 205  
 — спектроскопии в случае дискретного спектра 217  
 — — — непрерывного спектра 203  
**Задачи компьютерной томографии** 19  
 — с неполными данными 37  
**Закон Бера** (Бугера–Ламберта–Бера) 26  
**Зоигматография** 41, 62  
  
**Измерительное устройство** (ИЗУ) 16  
**Изображение** (image) 13, 68  
 — бинарное (черно-белое, двоичное, logical, bw) 70  
 — дефокусированное (out-of-focus, defocused) 156  
 — зашумленное (noisy) 117, 126, 130  
 — индексированное (цветное палитровое, ind) 70  
 — истинное (неискаженное) 13  
 — модельное (фантом) 178  
 — полутоновое (серое, gray, grayscale) 70  
 — портретное 125, 191  
 — реальное 151  
 — смазанное (smeared, blurred) 13, 15, 117, 126, 130  
 — — под углом 139  
 — текстовое 101, 169  
 — цветное (полноцветное, RGB) 70, 146, 170, 180, 194  
**Иконика** 69  
  
**Импульс высокочастотный** (ВЧ) 51  
**Импульсная последовательность** (ИП) 52  
 — реакция фильтра  $p(t)$  29  
**Импульсные последовательности** 56  
**Импульсный метод** 51  
**Импульсы Карра–Парселла** 55  
 $\pi/2$ -импульс 51  
**Инверсная фильтрация** 86, 172  
**Индукция магнитного поля**  $B$  41, 49  
**Интегральное уравнение** (ИУ) 15, 77, 159  
 — — Вольтерра I рода 77  
 — — двухмерное 188  
 — — неклассическое 77  
 — — нестандартное 77  
 — — Радона 27  
 — — сингулярное 33  
 — — Фредгольма I рода 206  
 — — — — с общим ядром двухмерное 162  
 — — — — — типа свертки 79  
 — — — — — — двухмерное 33, 79, 160  
 — — — — — — на полуоси 206  
**Интенсивность** 26  
**Искажения изображений** 69  
 — — типа "звоны" 84, 113, 168  
**Искомая функция** 16, 33  
**Исходный процесс** 11, 15  
**Итеративная регуляризация** 188  
**Итерации** (последовательные приближения) 189  
  
**Квантование изображения** 68  
**Классическая тихоновская регуляризация** 92  
**Классы чисел** (uint8, uint16, double, logical и др.) 70  
**Компьютерная томография** (КТ) 19  
**Константа** (параметр)  $K$  93, 176  
**Конус прецессии** 42  
**Корректность и некорректность** по Адамару 91  
**Космический телескоп "Хаббл"** (КТХ) 73  
 **$k$ -пространство** 58  
**Критерий Рэля** 227

- Ларморова прецессия 41
- Локальные поля 46
- Магнитное поле (градиентное, переменное, поляризирующее, постоянное) 12
- Магнитный момент протона  $\mu$  41
- — электрона 39
- — ядра 41
- Математическая схема моделирования измерений и обработки 16
- формулировка задачи спектроскопии 205
- Математическое описание задачи спектроскопии в случае непрерывного спектра 203
- Матрица единичная 208
- интенсивностей 68
- сенсоров 68
- транспонированная 92, 208
- Медианная фильтрация 117
- Метод адаптивной фильтрации Винера 117
- двусторонней полной вариации (bilateral total variation method – BTV method) 92
- двухмерного ПФ с регуляризацией Тихонова 33, 176
- дифференцирования 78
- импульсный 51
- инверсной фильтрации 86, 172
- итеративной регуляризации (итераций) Фридмана 188
- итераций с квадратурами 186
- квадратур (конечных сумм) и регуляризации Тихонова 89, 207
- — — — — вариант 3 90, 108
- — — — — вариант 4 90, 108, 114
- — — — — вариант 5 90
- Кумара–Велти–Эрнста 63
- локальной регуляризации Арсенина 36
- оптимальной фильтрации Винера 93, 175
- параметрической фильтрации Винера 93, 122, 176
- преобразования Фурье (ПФ) 28, 86, 172
- — — с регуляризацией Тихонова 88
- — — — — вариант 1 89, 103
- — — — — вариант 2 89, 106
- псевдоинверсной фильтрации 87, 175
- регуляризации Тихонова 34, 88, 91, 176, 208
- — — для уравнения типа свертки 88, 176
- свертки и обратной проекции 29
- фильтрации Винера 93
- Методика Хоулта 62
- Методы реконструкции ЯМР-изображений 64
- решения уравнения Радона 27
- Механический момент (спин) протона 41
- — — электрона 40
- Минимизация функционала 176
- Множество (совокупность) интегральных уравнений 15
- Модуль спектра 52
- МРТ (магнитно-резонансная томография) 39
- Н**апряженность магнитного поля  $H$  41
- Начальное приближение решения 189
- Недоопределенная СЛАУ 84, 91
- Нейтрон  $n$  41
- Некорректная задача (ill-posed problem) 27, 78, 87, 91, 174, 206
- Некорректность 91
- Неоднородность полей  $\Delta B_0$ ,  $\Delta G_x$ ,  $\Delta G_y$  46, 65
- Неустойчивость решения 35
- Нормальное псевдорешение 93
- Области применения РКТ** 37
- — спектрального анализа 201
- — ЯМР-томографии 66
- Область  $F$**  дефокусированного изображения  $g$  162
- исходного изображения (FOV – field of view) 82, 162

- Обработка изображений (image processing) 69
- реального смазанного изображения 151
  - цветных изображений 146
  - — — (два способа) 146
- Обратная задача (inverse problem) 15, 76, 86, 141, 143
- Обратное преобразование Фурье (ОПФ) 28, 64, 172
- Обратные задачи первого и второго типов 18
- — спектроскопии 15, 199
- Объекты спектроскопии 68
- Ограничения на решение 219
- Оператор 16
- обратный 17
- Операторное уравнение 16
- Оптическая передаточная функция (ОПФ) или optical transfer function (OTF)  $H(\omega)$  87, 173
- Оптические спектральные приборы 199
- Ориентация магнитных моментов ядер 41
- Относительная погрешность решения  $\sigma_{\text{rel}}$  214
- Относительное СКО  $\sigma_{\text{rel}}$  99, 100, 194
- Оцифровка изображения 68
- Ошибка — см. Погрешность
- Пакет Image Processing Toolbox (ИРТ)** 69
- Параметр  $y$  15
- (константа)  $K$  93, 176
  - регуляризации  $\alpha$  34, 88, 99, 209
- Переопределенная СЛАУ 91
- Переход вектора  $\mathbf{M}$  из начального положения в конечное 47, 49
- ПЗС 68
- ПЗС-матрица 68
- Пиксель 68
- Поворот изображения 140
- Поглощение 26
- энергии ядром 40
- Погрешность решения 214
- функции рассеяния точки 117
- Подавление высоких Фурье-частот  $\omega$  35
- Позитив и негатив 101
- Поле магнитное постоянное 45
- Поля градиентные  $G_x, G_y, G_z$  56
- локальные 46
- Помеха 16
- Портретное изображение 125
- Порядок регуляризации  $p$  34, 88
- Правая часть 15, 16, 33
- Преобразование Радона 26
- уравнения к уравнению типа свертки 78
  - Фурье (ПФ) 11, 28, 34
  - — быстрое (БПФ) 35, 87
  - — двухмерное 28, 86
  - — — с масштабированием 64
  - — дискретное (ДПФ) 35
  - — непрерывное (НПФ) 35
  - — обратное (ОПФ) 28, 64, 172
  - — трехмерное 65
- Прецессия вектора  $\mathbf{M}$  46
- магнитного момента протона 43
- Приведение основного уравнения к стандартной форме 160
- уравнения Радона к уравнению Фредгольма I рода 32
- Прием (способ) поворота изображения 140
- — размытия краев изображения 84, 167
  - — усечения изображения 82, 163
- Приемник 22
- Пример исходный (оригинал)  $P$  211
- модельный  $Q$  211
- Примеры восстановления (реконструкции) изображений 101, 125, 178
- задач спектроскопии 211, 221
  - искаженных изображений 71, 76, 158, 161, 178
  - обратных задач 15
  - прикладных задач 11, 15
  - прямых задач 11
  - реконструкции рентгеновских изображений 35
  - — ЯМР-изображений 66
- Программирование в системе MatLab (некоторые особенности) 225



- Программы головные в системе MatLab  
96, 98
- Прозрачность 26
- Пространства Соболева  $W_2^1, W_2^2, W_2^n$  92
- Пространственное кодирование частоты  
резонанса 58
- Пространство  $L_1$  92
- Пространство  $L_2$  92, 96
- $k$ -пространство 58
- Протон 41
- Процесс входной 14, 16  
— выходной 15, 16
- Процессы релаксации 46
- Прямые и обратные задачи 11, 74, 75,  
76, 126, 162
- Псевдоинверсная фильтрация 87, 175
- Псевдообратный оператор 210
- Псевдорешение 17, 93
- Пять поколений рентгеновских  
томографов 22
- Рабочий объем томографа 66**
- Размытие краев изображения 84
- Разрешающая способность измерительно-  
го устройства 230  
— спектрометра 16  
— — ЯМР-изображений 65
- Распределение интенсивности вдоль  
истинного (неискаженного)  
изображения 13, 15  
— — — смазанного изображения 13, 15  
— — (энергии) по спектру 14
- Расстройка по частоте  $\Delta\omega = \omega_0 - \omega$  44
- Расфазирование спинов (потеря резонан-  
са) 46
- Регуляризованное решение 34, 208
- Регуляризованный спектр решения 34, 88
- Редукционная проблема Рэля 16, 229
- Редукция измерений к идеальному изме-  
рительному устройству 229
- Резонанс ( $\omega_0 = \omega$ ) 45
- Реконструкция (восстановление) дефоку-  
сированных изображений 156,  
172, 178, 180
- зашумленных изображений 94  
— изображений 69  
— искаженных изображений 68  
— рентгеновского изображения 27  
— смазанных изображений 74, 86, 94  
— ЯМР-изображений 58, 62
- Релаксация 46  
— спин-решетчатая (продольная) 46  
— спин-спиновая (поперечная) 46
- Рентгеновская компьютерная  
томография (РКТ) 21
- Рентгеновский томограф (5 поколений) 22
- Рентгенография 21
- Решение в пространстве  $S_{\pi/\Delta x}$  87  
— неустойчивое 35  
— регуляризованное 34, 208  
— точное 93  
— устойчивое 93
- РКТ-сканер 25
- Сглаживающая фильтрация методом на-  
именьших квадратов со связью 89**
- Сглаживающий функционал Тихонова  
92, 176
- Сдвиг (смещение) 76  
— изображения 150
- Сенсор 68
- Сечение и слой 60  
 $z$ -сечения 13
- Сигнал дисперсии (рассеяния) 52  
— поглощения 52  
— спада свободной индукции (ССИ) 51
- Сингулярное ИУ 33
- Система линейно-нелинейных уравнений  
(СЛНУ) 219  
— линейных алгебраических уравнений  
(СЛАУ) 91, 91, 208  
— — — — недоопределенная 84, 91  
— — — — определенная 91  
— — — — переопределенная 91  
— нелинейных уравнений (СНУ) 219  
— программирования MatLab 81
- Системы наблюдений 68
- Сканирующая функция 16

- Слой 60
- Смаз (сдвиг, смещение) 13, 102
- Смазанное изображение 13, 15
- Смазывание изображения (image smearing) 13
- Собственная m-функция (m-файл) 81
- — defocusing.m 169
  - — desmearingf.m 89, 96
  - — desmearingq.m 93, 97
  - — error\_rel.m 214
  - — medfilt2c.m 180
  - — normL2.m 96
  - — normnoise.m 95
  - — refocusingF.m 190
  - — refocusingT.m 177
  - — rmsd.m 100, 151
  - — smearing.m 84, 84, 95
  - — Tikh.m 208
  - — wiener2c.m 180
- Совмещение восстановленного и точного изображений 151
- Совокупность (множество) интегральных уравнений 15, 77
- Соглашение о координатах 70
- Спектр (дискретный, измеренный, истинный, непрерывный, полосатый, точный, экспериментальный) 14, 15, 173, 199, 204, 205, 212, 217
- импульса 61
  - процесса 11
  - решения 173
  - узкополосный 201
  - Фурье 34
  - широкополосный 201
  - ЯМР-сигнала 52
- Спектральная плотность мощности (помехи, решения) 93, 175
- чувствительность (СЧ) спектрометра 16, 203
- Спектральные приборы (спектрографы, спектроскопы, спектрометры, спектрофотометры) 199
- Спектральный анализ 199
- Спектрометрия (узкополосная, широкополосная) 201
- Спин (механический момент) электрона 40
- Спин-эхо-ИП (SE, spin echo) 54
- — — Карра-Парселла (КП) 55
- Способ модельных (эталонных) примеров (способ моделирования, метод вычислительных экспериментов, способ псевдообратного оператора) 209
- Способы выбора параметра регуляризации  $\alpha$  99, 209
- Сравнение методов Тихонова и Винера 93, 125, 135, 176
- Среднеквадратическая погрешность 99
- Среднеквадратическое отклонение (СКО) 99, 175
- Стабилизатор  $\Omega[w]$  92
- Сферическая абберрация 72
- Схема дискретного спектра 219
- для понижения эффекта Гиббса 84
  - получения дефокусированного изображения 157
  - — смазанного изображения 75
  - расчета дефокусированного изображения 165, 166, 167, 168
  - рентгеновского томографа 22
- Схемы измерений и обработки 16
- Текстовое изображение 101
- Теорема Брейсуэлла о сечении спектра 27
- Котельникова 30
  - о свертке 86
- Тесла 39, 49
- Техническая идея РТ 21
- Типичная рентгенограмма 21
- Типы абберраций 72
- изображений 69
  - искажений изображений 68, 69
  - спектрального анализа 201
  - спектров 199
- Томограммы 36
- Томограф рентгеновский (5 поколений) 22

- ядерно-магнитно-резонансный 66
- Томография 19
- магнитно-резонансная (МРТ) 39
- рентгеновская компьютерная (РКТ) 21
- ядерно-магнитно-резонансная 39
- Тонкая структура импульса  $p(t)$  60
- У**гловая скорость ларморовой прецессии
  - $\omega_0$  42, 49
- Угол нутации Эйлера 45
- смаза 102
- Уравнение гармонических колебаний 42
- дифференциальное Блоха 45
- — Лармора 41
- интегральное (ИУ) 15, 33
- — Вольтерра 77
- — Радона 26
- — Фредгольма 206
- операторное 16
- Уравнения Блоха 45
- Устройство регистрации изображений (УРИ) 13, 69
- Ф**айл (m-файл) 11, 81
- Физико-техническая схема измерений и обработки 16
- Фильтр Рамачандрана–Лакшминараяна 31
- Троицкого модифицированный 31
- Шеппа–Логана 31, 36
- Фильтрация инверсная 86, 172
- медианная 117
- методом минимизации сглаживающего функционала со связью (метод регуляризации Тихонова) 89
- псевдоинверсная 87, 175
- Формула Эйлера 30
- Функции внешние, встроенные (внутренние), собственные 81
- Функция аппаратная (АФ) 16
- искомая 16, 33
- рассеяния точки (ФРТ) (Point Spread Function – PSF) 16, 79, 157, 160, 164
- финитная 82
- m-функция 81
- Х**арактеристика направленности (ХН) антенны 16
- Ц**ветное изображение, см. Изображение цветное
- Ч**астота вращения полей  $\omega$  44
- Найквиста 30
- нутации 44
- прецессии 44
- Фурье 11
- Ш**ум 16, 117, 126, 130
- Шумовая компонента детерминированная 204
- — случайная 204
- Э**кстраполяция значений  $w$  за границу  $B$  81
- Электромагнитный ЯМР-сигнал 12
- Энергетический спектр 175
- Эффект Гиббса (искажения типа "звоны", ложные волны) 84, 113
- Доплера 200
- Зеемана 200
- Штарка 200
- Эхо-сигнал 45
- Я**дерная магнитная восприимчивость  $\chi$  43
- Ядерно-магнитно-резонансная томография 39
- Ядерный спин 43
- Ядро интегрального уравнения 15, 33, 160
- ЯМР (ядерный магнитный резонанс) 39, 45
- ЯМР-изображение (томограмма) 40
- ЯМР-сигнал 12, 15, 45, 51, 51
- ЯМР-томограф 66
- ЯМР-томография (МРТ) 39



В. С. Сизиков

# Обратные прикладные задачи



# и MATLAB



$$Au_y = \int_A(x, \xi) \omega_y(\xi) d\xi = g_y(x), \quad c \leq x \leq d$$

function w\_alpha = Desmearingq(g,Delta,theta,  
invariant,alpha,occasion,

