



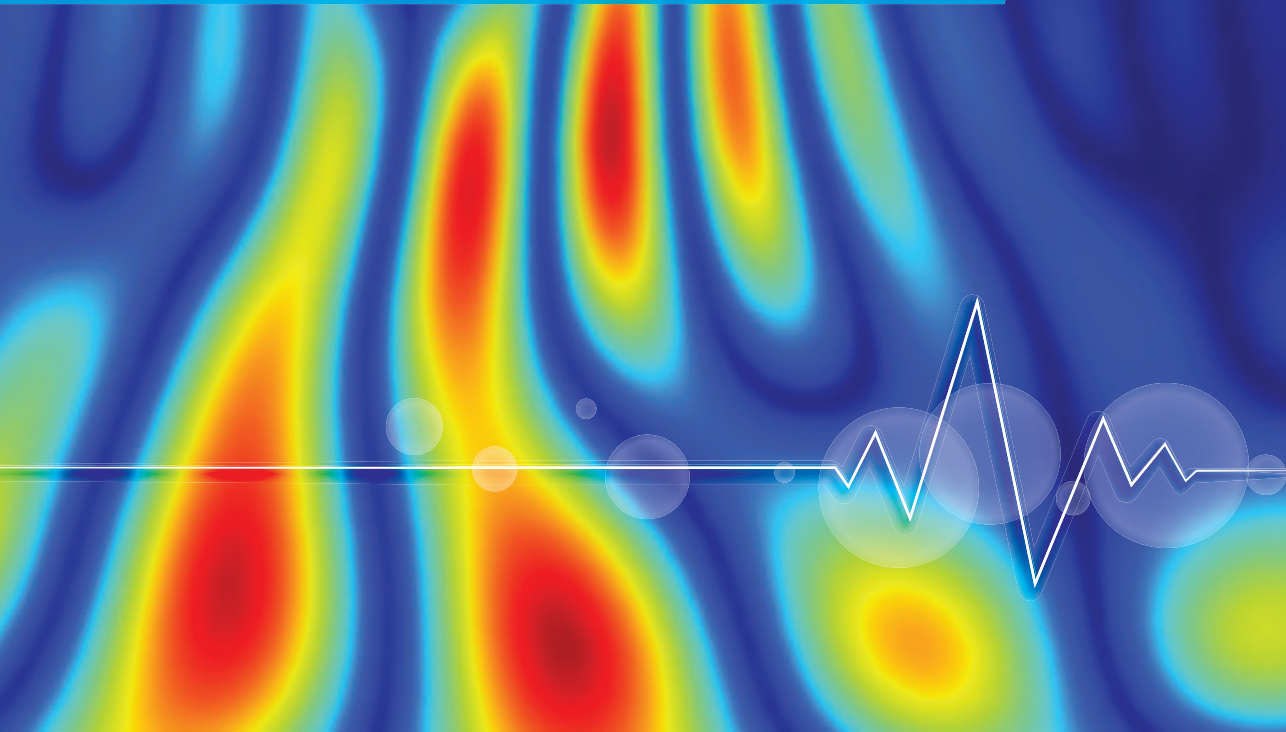
Уральский
федеральный
университет

имени первого Президента
России Б.Н.Ельцина

Институт радиоэлектроники
и информационных
технологий — РТФ

БИОМЕДИЦИНСКИЕ СИГНАЛЫ И ИЗОБРАЖЕНИЯ В ЦИФРОВОМ ЗДРАВООХРАНЕНИИ: ХРАНЕНИЕ, ОБРАБОТКА И АНАЛИЗ

Учебное пособие



Министерство науки и высшего образования
Российской Федерации
Уральский федеральный университет
имени первого Президента России Б. Н. Ельцина

Биомедицинские сигналы и изображения в цифровом здравоохранении: хранение, обработка и анализ

Учебное пособие

Под общей редакцией
доктора технических наук В. С. Кубланова

Рекомендовано методическим советом
Уральского федерального университета
для студентов вуза, обучающихся
по направлениям подготовки
09.04.02 — Информационные системы и технологии,
12.04.04 — Биотехнические системы и технологии,
27.04.03 — Системный анализ и управление

Екатеринбург
Издательство Уральского университета
2020

УДК 615.84:621.39:004.3(075.8)

ББК 53.4я73+32.8я73

Б63

Авторы:

В. С. Кубланов (гл. 1, 2), А. Ю. Долганов (гл. 2, 3 (перевод), 4), В. Б. Костоусов (гл. 6), А. П. Немирко (гл. 5), Л. А. Манило (гл. 5), Т. С. Петренко (гл. 1), Н. Gamboa (гл. 3), J. Rodrigues (гл. 3)

Рецензенты:

заведующий кафедрой инфокоммуникационных технологий Южно-Уральского государственного университета (научно-исследовательский университет), д-р техн. наук, доц. С. Н. Даровских;
лаборатория морфологии и биохимии института иммунологии и физиологии Уральского отделения РАН (замдиректора ИИиФ УрО РАН д-р биол. наук, доц. И. Г. Данилова)

Биомедицинские сигналы и изображения в цифровом здравоохранении: хранение, обработка и анализ : учебное пособие / В. С. Кубланов, А. Ю. Долганов, В. Б. Костоусов [и др.] ; [под общ. ред. В. С. Кубланова] ; Мин-во науки и высш. образования РФ. — Екатеринбург : Изд-во Урал. ун-та, 2020. — 240 с.

ISBN 978-5-7996-2990-8

В учебном пособии изложены современные практические рекомендации по математическим методам и алгоритмам анализа биомедицинских сигналов и изображений.

Методы цифровой обработки биомедицинских сигналов и изображений излагаются с учетом специфических особенностей сигналов биологического происхождения и рассматриваются на конкретных примерах, иллюстрирующих их практическое применение. Основное внимание уделено работе с существующими базами данных биомедицинских сигналов, а также методикам проведения собственных функциональных исследований.

Учебное пособие ориентировано на преподавателей, аспирантов и студентов-магистрантов, обучающихся по специальностям, связанным с цифровой обработкой сигналов и изображений, интеллектуальными информационными системами и функциональной диагностикой. Пособие будет полезно также врачам, интересующимся современными методами компьютерного анализа биомедицинских сигналов и изображений и методами машинного обучения в задачах классификации данных.

Работа поддержана Постановлением Правительства Российской Федерации № 211, договор 02.А03.21.0006.

Библиогр.: 124 назв. Табл. 1. Рис. 72.

УДК 615.84:621.39:004.3(075.8)

ББК 53.4я73+32.8я73

ISBN 978-5-7996-2990-8

© Уральский федеральный
университет, 2020

Оглавление

Список сокращений	5
Введение	7
Глава 1. Обзор общих принципов функционирования организма	12
1.1. Механизмы регуляции. Функции организма	16
1.2. Высшая нервная и психическая деятельность.....	18
1.3. Системный принцип регуляции функций организма.....	22
Контрольные вопросы к главе 1	27
Глава 2. Биомедицинские сигналы в задачах функциональной диагностики: принципы формирования исследований и их организация	28
2.1. Электрокардиограмма	31
2.2. Электроэнцефалограмма	36
2.3. Сигналы кожно-гальванической реакции	38
2.4. Стабилографический сигнал	40
2.5. Речевой сигнал	43
2.6. Биосигналы функциональной диагностики второй группы.....	45
2.7. Организация функциональных исследований	47
Практическое задание к главе 2	54
Контрольные вопросы к главе 2	54
Глава 3. Хранение биомедицинских сигналов	56
3.1. Форматы биосигналов	57
3.2. Базы данных биосигналов	71
3.3. API для баз данных и анализа биосигналов	74
Практическое задание к главе 3	77
Контрольные вопросы к главе 3	78
Глава 4. Расчет параметров биомедицинских сигналов	79
4.1. До обработки сигналов	79
4.2. Предобработка сигналов	82
Практическое задание к п. 4.1 и 4.2	86

4.3. Статистические параметры	86
Практическое задание к п. 4.3	89
4.4. Гистограммы распределения.....	89
Практическое задание к п. 4.4	92
4.5. Спектральные показатели (Фурье-анализ).....	92
Практическое задание к п. 4.5	95
4.6. Спектральные показатели (вейвлет-преобразование)	96
Практическое задание к п. 4.6	99
4.7. Нелинейные оценки	100
Практическое задание к п. 4.7	105
Контрольные вопросы к главе 4	105

Глава 5. Применение методов машинного обучения для анализа

биомедицинских сигналов	106
5.1. Методы классификации	107
5.2. Метод главных компонент	116
Практическое задание к п. 5.2	120
5.3. Линейный дискриминантный анализ.....	121
Практическое задание к п. 5.3	125
5.4. Машинное обучение нейрона	126
Практическое задание к п. 5.4	132
Контрольные вопросы к главе 5	132

Глава 6. Обработка изображений в биомедицинских задачах

6.1. Базовые операции.....	136
Практическое задание к п. 6.1	158
6.2. Методы улучшения изображений:	158
Практическое задание к п. 6.2	167
6.3. Восстановление изображений.....	167
Практическое задание к п. 6.3	182
6.4. Дескрипторы особенностей	182
Практическое задание к п. 6.4	195
6.5. Обработка видео последовательности	195
Практическое задание к п. 6.5	215
6.6. Анализ видеоизображения для оценки сердечного ритма	215
Практическое задание к п. 6.6	228
Контрольные вопросы к главе 6	228

Список библиографических ссылок.....	230
---	------------

Список сокращений

АВ	— атриовентрикулярный (узел)
АД	— артериальное давление
АРД	— акцептор результата действия
ВНД	— высшая нервная деятельность
ВНС	— вегетативная нервная система
ВОЗ	— Всемирная организация здравоохранения
ВПр	— вегетативный показатель ритма
ВР	— вегетативное равновесие
ВСР	— вариабельность сердечного ритма
ДПВ	— длительность пульсовой волны
ДПФ	— дискретное преобразование Фурье
ДФ	— дискриминантная функция
ИАП	— индекс активации подкорковых нервных центров
ИВР	— индекс вегетативного равновесия
ИН	— индекс напряжения регуляторных систем
КВК	— коэффициент взаимной корреляции
КГР	— кожно-гальваническая реакция
КИХ	— конечная импульсная характеристика
ЛГ	— лапласская гауссиана
ЛДФ	— линейная дискриминантная функция
ЛЖ	— левый желудочек
ЛП	— левое предсердие
МА	— мерцательная аритмия
НР	— нормальный ритм
ПАПр	— показатель адекватности процессов регуляции
ПЖ	— правый желудочек
ПЛБ	— порогового логического блока
ПП	— правое предсердие
ПрВ	— плотность распределения вероятностей

СА	— синоартриальный (узел)
ФВК	— функция взаимной корреляции
ФВЧ	— фильтр высоких частот
ФНЧ	— фильтр низких частот
ФР	— функция распределения
ЦД	— центр давления
ЦНС	— центральная нервная система
ЧСС	— частота сердечных сокращений
ЧЭ	— частая экстрасистолия
ЭКГ	— электрокардиограмма
ЭКоГ	— электрокортикограмма
ЭМГ	— электромиография
ЭОГ	— электроокулография
ЭЭГ	— электроэнцефалограмма
BRIEF	— binary Robust Independent Elementary Features
FAST	— feature from Accelerated Segment Test
HOG	— histogram of oriented gradients
MLDB	— modified Local Difference Binary
ORB	— oriented FAST and Rotated BRIEF
rPPG	— remote photoplethysmography
SCL	— skin conductivity level
SCR	— skin conductivity response
SIFT	— scale Invariant Feature Transform
SURF	— speeded-UP Robust Features

Введение

Изменения условий жизни популяции в XXI веке во многом определяются достижениями науки и техники, которые способствуют увеличению продолжительности жизни человека и замедлению его старения. Эти процессы обуславливаются успехами в профилактике, ранним выявлением и лечением заболеваний, особенно детей и пожилых людей.

Следует отметить, что эти изменения наблюдаются на фоне возрастающего темпа жизни, избытка электронной информации, дефицита времени, снижения физической активности, гиподинамии, урбанизации, увеличения частоты техногенно-экологических катастроф, военных и социальных конфликтов и, по существу, охватили все сферы производственной и социальной деятельности человека. В этих условиях создаются предпосылки для нарастания психоэмоциональных перегрузок на человека и ухудшения его адаптации к изменениям окружающей среды. Это отражается на изменении структуры и причин смертности.

Основными причинами смерти по-прежнему являются болезни системы кровообращения и злокачественные новообразования. Цереброваскулярные заболевания отодвинулись на четвертое место, а третье место заняли хронические болезни нижних дыхательных путей. Далее следуют нейродегенеративные заболевания, которые сопровождаются когнитивными и моторными нарушениями. Другие хронические заболевания и нарушения здоровья в меньшей степени ограничивают деятельность пожилых людей, хотя в старшей возрастной группе роль нарушений зрения и слуха, а также старческого слабоумия резко возрастает [1].

Почему именно в настоящее время эффективное внедрение цифровых технологий в сферу здравоохранения является важнейшим элементом создания крепких и слаженно работающих систем здравоохранения,

а также расширения прав и возможностей пациентов в контексте перехода к помощи, ориентированной на нужды людей?

Охрана здоровья населения является одним из ведущих компонентов мирового развития. В материалах ТАСС от 22.09.2019 (<https://tass.ru/obschestvo>), Всемирная организация здравоохранения (ВОЗ) считает, что странам следует увеличить ежегодные капиталовложения в первичную медицинскую помощь не менее чем на 1 % валового внутреннего продукта. Такие инвестиции в целях устойчивого развития позволят человечеству сделать значительный шаг на пути к выполнению поставленной Организацией объединенных наций задачи всеобщего охвата к 2030 году населения медико-санитарными услугами. Для эффективного управления этими ресурсами как никогда актуальной становится развитие и внедрение методов цифрового управления экономикой. Примером такого подхода являются США, где объем инвестиций в цифровое здравоохранение увеличивается каждый год в 1,5 раза.

В России цифровое управление экономикой поддерживается на федеральном уровне: целевая программа «Цифровая экономика РФ» определена распоряжением Правительства от 28 июля 2017 г. № 1632-р и предусматривает создание отраслевых цифровых платформ по различным направлениям экономической деятельности. Этой программой определены цели, задачи, направления и сроки реализации основных мер государственной политики по созданию необходимых условий для развития в России цифровой экономики, в которой данные в цифровом виде являются ключевым фактором производства во всех сферах социально-экономической деятельности. В полной мере это относится и к здравоохранению.

Для отечественного здравоохранения открытие направления «цифровая медицина» является реальным шансом занять лидирующие позиции в структуре экономического роста страны: как в федеральных лечебно-профилактических учреждениях, так и в частном медицинском бизнесе открываются новые возможности для развития информационных технологий, обеспечивающих приближение квалифицированной медицинской помощи непосредственно к пациенту.

Сегодня как никогда ключевая информация здравоохранения, на основании которой будут приниматься решения об оказании клинической помощи, должна быть доступна в нужное время и нужном месте. Здесь цифровому здравоохранению отводится весомая роль в выполне-

нии ключевых приоритетных задач общественного здравоохранения, сформулированных в европейской политике здравоохранения «Здоровье-2020», которая была утверждена государствами-членами в 2012 г.

Политика «Здоровье-2020» предполагает создание крепких межсекторальных механизмов для минимизации различных факторов риска и воздействия на детерминанты здоровья для сокращения неравенств в здоровье и значительных показателей здоровья и благополучия населения. Повестка дня ООН в области устойчивого развития на период до 2030 г. продолжает и расширяет концепцию политики «Здоровье-2020», предполагая укрепление потенциала государств-членов для достижения более высоких, справедливых и стабильных показателей здоровья и благополучия на всех этапах жизненного пути.

Цифровые решения могут способствовать достижению связанных со здоровьем целей устойчивого развития и обеспечению всеобщего охвата услугами здравоохранения путем революционных преобразований в предоставлении помощи и реформирования процессов, лежащих в основе систем здравоохранения.

Признавая потенциал цифровых технологий для укрепления национальных систем здравоохранения и выполнения названных выше целей, государства-члены ВОЗ приняли на Семьдесят первой сессии Всемирной ассамблеи здравоохранения в мае 2018 г. резолюцию WHA71.7, которая определяет роль цифрового здравоохранения для укрепления системы здравоохранения и в качестве средства содействия справедливому, приемлемому по стоимости и всеобщему доступу к услугам в области здравоохранения для всех.

Таким образом, функциональная неполноценность становится ключевым фактором здоровья людей, особенно пожилого возраста, а мониторинг здоровья — обязательным элементом формирования полноценного образа жизни. В этом случае обеспечение системы мониторингирования становится все более сложной задачей. Необходимость согласованности национальных политик развития здравоохранения в каждом государстве побудили ВОЗ разработать предложения для формирования стратегии национальных программ здравоохранения [2].

Одним из основных положений этих стратегий является обеспечение всеобщего охвата услугами здравоохранения: все люди должны иметь доступ к необходимым медицинским услугам по профилактике, лечению и реабилитации, достаточного качества, чтобы быть эффективными. При этом люди не должны испытывать финансовых

трудностей при оплате этих услуг, что влечет за собой сокращение разрыва между доступом, потребностью в услугах и их использованием, улучшением качества и улучшением финансовой защиты.

Применение стандартов обеспечивает большую юридическую защиту врачей, улучшение организации медицинской помощи и минимизацию материальных затрат. В настоящее время стандартизация во всем мире признается одной из главных составляющих системы управления качеством в здравоохранении.

Противоречит ли это принципам лечения больного, а не болезни, заложенных еще трудами Гиппократов? Ведь «индивидуализация лечения всегда служила основой российской терапевтической школы: целью исследования врачей была не только болезнь, но и изучение состояния больного организма во всей целостности его физиологических и биологических проявлений» [3].

Для того чтобы эти два подхода дополняли друг друга и способствовали повышению эффективности здравоохранения, необходимо, чтобы стандарты не были обязательными догмами, которым нужно слепо следовать, а рассматривались как рекомендации, в соответствии с которыми врач может выбрать и оценить тактику ведения больного. Такой подход более точно отражает реальность медицины, где не должно быть ничего абсолютного. В этом случае больной не теряет своей уникальности и каждая клиническая ситуация будет рассматриваться не как «типичная», а с учетом присущих пациенту индивидуальных особенностей. В этом свете актуальным представляется постулат последних Европейских рекомендаций по диагностике и лечению артериальной гипертензии, в которых указывается, что данные рекомендации носят характер «образовательных, а не регламентирующих лечение конкретного пациента, которое можно существенно варьировать в зависимости от личных, медицинских и культурных условий», что может позволить врачу принять решение, отличное от прописанных в стандартах [4]. Поэтому можно рассматривать эту проблему с позиции управления биотехническими системами, в которых индивидуальные особенности пациента будут анализироваться в системе данных, формируемых по нозологическим признакам. Ключевыми атрибутами исследования в этом случае являются правильное планирование, выбор интерфейса для сбора данных и их объективная интерпретация. Планирование исследования обеспечивает стандартизацию дизайна исследования биотехнических си-

.....

стем и включает для каждого объекта исследования определенный тип функционально-нагрузочных проб.

В качестве объектов исследования в этом случае выступают функциональные системы, которые являются временным функциональным объединением нервных центров различных органов и систем организма для достижения метаболических, гомеостатических, поведенческих или социальных результатов [5]. В этих исследованиях на объект воздействуют с помощью внешних факторов, провоцирующих патофизиологические состояния. Как следствие, в результате этого начинает проявляться скрытая или труднодоступная для обнаружения в условиях покоя патология. В тех случаях, когда эта патология как нозологическая форма известна до исследования, с помощью функциональных проб удастся определить степень ее выраженности или компенсаторные возможности регуляторных систем объекта исследования.

Для обеспечения достоверности и воспроизводимости результатов исследований необходимо, чтобы они не зависели от места их проведения и личности исследователя, а также регистрировали сопутствующие данные, такие как время исследования, условия исследования (сделаны они до или после еды, во время менструального цикла и т. д.).

Вопросы организации функциональных исследований, сбора данных и методов их обработки и анализа рассматриваются в материалах настоящего учебного пособия.

Для обработки биомедицинских сигналов и изображений выбран язык *Python*. Этот язык имеет открытый доступ. Следует отметить, что для языка *Python* реализовано большое число библиотек с широким спектром применений от вычислений при обработке отдельных сигналов и изображений до алгоритмов машинного обучения.

Полные версии листингов программ, используемых в учебном пособии, доступны в онлайн-хостинге репозитория [GitHub: Biomedical-Signals-and-Images](#).

Глава 1.

Обзор общих принципов функционирования организма*

Организм является открытой саморегулирующейся системой, которая обеспечивает непрерывный обмен веществ и энергии с окружающей средой и способна адаптироваться к меняющимся условиям этой среды в различных условиях жизнедеятельности [6]. Принято различать физиологические и функциональные системы организма.

Физиологическая система — это наследственно закреплённая совокупность органов и тканей, выполняющая определённую функцию. К ним относятся: покровная система (кожа), нервная, эндокринная, иммунная, сердечно-сосудистая системы, а также системы крови, дыхания, пищеварения, выделения, воспроизведения. Согласно исследованию П. К. Анохина [7], функциональной системой является динамическая совокупность отдельных органов и физиологических систем, формирующаяся для достижения полезного для организма приспособительного результата. При этом *гомеостатические функциональные системы* поддерживают оптимальные физиологические показатели, обеспечивающие жизнедеятельность организма в разных условиях, а *поведенческие функциональные системы* обеспечивают достижение полезного результата в виде какой-то цели удовлетворения биологических, социальных и бытовых потребностей, что, в свою очередь, способствует поддержанию показателей на оптимальном уровне. Их число огромно.

Например, для восстановления нормальной величины снизившегося артериального давления может оказаться достаточным учащения серд-

* Авторы главы 1: Т. С. Петренко, В. С. Кубланов (Уральский федеральный университет, г. Екатеринбург)

цебиений и сужения кровеносных сосудов с помощью нервной системы. Однако после существенного изменения артериального давления для его восстановления до нормы формируется более сложная функциональная система, включающая не только сердечно-сосудистую и нервную системы, но и системы выделения (почки выделяют больше или меньше воды), эндокринную, системы крови и пищеварения [8].

В общем случае функция (от лат. *functio* — деятельность) определяется специфической деятельностью клеток, органов и систем по обеспечению жизнедеятельности целого организма. При этом функцией сердца является нагнетание крови в артериальную систему, функцией эндокринных желез — выработка гормонов, функцией желудочно-кишечного тракта — обеспечение поступления питательных веществ в кровь, что достигается с помощью механической и химической обработки пищевых веществ и всасывания в кровь и лимфу продуктов их гидролиза и т. д. [9].

Одним из главных условий полноценного функционирования организма является поддержание стабильности внутренней среды и окружения. Это требует постоянного мониторинга и коррекции этих процессов при изменении внешних условий. Такое регулирование физиологических систем в организме называется *гомеостатической регуляцией* [10].

Гомеостатическая регуляция включает три части, или механизма: рецептор, центр управления и эффектор.

Рецептор получает информацию о том, что что-то в окружающей среде меняется. Центр управления получает и обрабатывает информацию от приемника этой информации. И, наконец, эффектор реагирует на команды центра управления либо противодействуя, либо усиливая стимул. Этот непрерывный процесс постоянно восстанавливает и поддерживает гомеостаз. Например, при регулировании температуры тела температурные рецепторы кожи передают информацию в мозг, который является центром управления, а эффектор — это наши кровеносные сосуды и потовые железы кожи.

Гомеостаз, таким образом, представляет собой некое синтетическое равновесие, поддержание которого обеспечивается ограничением флуктуаций внутренней среды при помощи положительных и отрицательных обратных связей.

Отрицательная обратная связь обеспечивает такую реакцию, при которой система изменяет вектор направления происходящих изме-

нений. Например, когда концентрация углекислого газа в организме человека увеличивается, легкие начинают работать более активно, увеличивая экстракцию углекислого газа из крови.

При *положительной обратной связи* происходит усиление происходящих изменений. Положительные обратные связи менее распространены в естественных системах, чем отрицательные, но имеют важное значение и обычно более мощные.

Генерация возрастающего потенциала действия в нервных волокнах является примером положительной обратной связи. Система свертывания крови реализуется через двенадцать каскадных реакций, последовательно усиливающих друг друга, приводя к превращению крови в твердое вещество. В процессе родов подобным образом происходит усиление секреции окситоцина, усиливающего сокращения матки.

Негативными последствиями положительных обратных связей является образование «порочных кругов», оказывающих негативное влияние на организм. Например, в случае некоторых инфекций повышается температура тела, возникающие при этом метаболические изменения «заставляют» гипоталамус (центр управления) поднимать температуру тела все выше и выше. В итоге это может закончиться смертельным исходом, так как при температуре 45 °С клеточные белки перестают работать.

Устойчивые системы требуют комбинаций обоих видов обратной связи. Как правило, с наступлением отклонения от гомеостатического состояния первыми включаются положительные обратные связи. Приближаясь к гомеостатическому равновесию, отрицательные обратные связи используются для «тонкой настройки». Создается состояние *метастабильности*, при котором гомеостаз поддерживается в фиксированных пределах, но как только эти пределы превышены, система может резко перейти к совершенно новому (и, возможно, менее желательному) состоянию гомеостаза [11].

Гомеостатические системы имеют сложную внутреннюю структуру, легко объединяются друг с другом, образуя новые функциональные системы. Открытость этих систем позволяет выбирать способ самоорганизации и коррекции необходимых переменных. Все это обеспечивает высокую надежность и *сверхстабильность* [6].

Условная схема модели работы гомеостаза представлена на рис. 1.1.

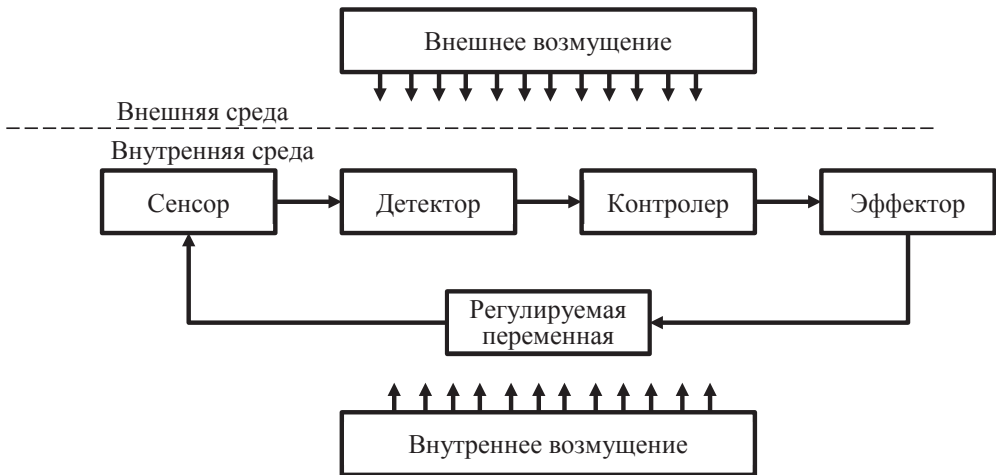


Рис. 1.1. Модель регуляции гомеостаза

Эта модель включает в себя пять обязательных компонентов, которые обеспечивают поддержание гомеостаза:

- 1) сенсор, который измеряет значение регулируемой переменной;
- 2) механизм установления «нормального диапазона» значений для регулируемой переменной;
- 3) детектор, который сравнивает сигнал, передаваемый сенсором, с нормальным значением: результатом этого сравнения может являться «сигнал ошибки», который передается на контроллер;
- 4) контроллер интерпретирует «сигнал ошибки» и определяет параметры выходного сигнала на эффектор;
- 5) эффекторами являются те элементы, которые определяют значение регулируемой переменной.

Адаптационные возможности гомеостаза существенно зависят от генетических особенностей индивидуума и образа жизни, включая питание, физическую активность, психологический фон, заботу о своем здоровье.

Каждая система организма способствует гомеостазу других систем и всего организма в целом. Ни одна система организма не работает в изоляции, и благополучие человека зависит от благополучия всех взаимодействующих систем. Нарушение в пределах одной системы обычно имеет последствия для остальных систем организма.

Различают несколько основных механизмов регуляции функций организма: нервный, гуморальный и миогенный.

1.1. Механизмы регуляции. Функции организма

Основным принципом работы нервной системы в контексте регуляции гомеостаза является передача нервного импульса к эффекторному звену, представленному в большинстве случаев мышечными волокнами и железами внутренней секреции. Таким образом, нервная система осуществляет либо стимулирующее, либо угнетающее воздействие на органы-мишени [12].

Нервная система состоит из двух основных частей: центральной нервной системы (ЦНС) и периферической нервной системы. Периферическая нервная система состоит из черепных и спинных нервов. Вегетативная (автономная) нервная система (ВНС) является частью периферической нервной системы и содержит двигательные нейроны, которые регулируют деятельность внутренних органов, желез внутренней и внешней секреции, кровеносных и лимфатических сосудов. ВНС играет ведущую роль в поддержании постоянства внутренней среды организма и в приспособительных реакциях всех позвоночных. Анатомически и функционально ВНС подразделяется на симпатический, парасимпатический и метасимпатический отделы.

Симпатические и парасимпатические отделы находятся под контролем коры больших полушарий и гипоталамических центров. При этом симпатическая нервная система усиливает обмен веществ, повышает возбудимость большинства тканей, мобилизует силы организма на активную деятельность и приводит к тем результатам, которые мы ассоциируем с экстремальными ситуациями, часто называемыми реакциями борьбы или бегства. парасимпатическая нервная система производит те эффекты, которые необходимы для нашего повседневного функционирования. Парасимпатическая система способствует восстановлению израсходованных запасов энергии, регулирует работу организма во время сна.

В контексте системной регуляции функций нервная система реализует два типа воздействия: пусковое и модулирующее. Примером пускового воздействия служит активация скелетных мышц по сигналу из двигательных зон коры больших полушарий головного мозга. Реализация механизмов регуляции гомеостаза происходит в основном путем модулирующего воздействия, т. е. угнетения или стимуляции уже работающих функций. Примером может служить регуляция сердеч-

ной деятельности. Причем симпатическая и парасимпатическая системы могут одновременно оказывать разнонаправленное действие, подстраивая (модулируя) таким образом функциональные характеристики к наиболее оптимальным для текущей окружающей обстановки [13].

Одним из базовых механизмов гомеостатической регуляции посредством нервной системы является рефлекс. Рефлекс представляет собой реакцию организма в ответ на раздражение сенсорных систем, осуществляемую под управлением нервной системы. Ответная реакция может быть в виде активации или торможения деятельности ткани или органа.

Каждый рефлекс осуществляется посредством рефлекторной дуги, состоящей из нескольких звеньев [6]:

- 1) первое звено — рецептор. Обеспечивает восприятие раздражителей, возникающих в результате возмущения внешней или внутренней среды. В рецепторах происходит трансформация энергии раздражителей в рецепторный потенциал, который накапливается в рефлексогенной зоне (зоне расположения однотипных рецепторов), а затем отправляется в виде электрического (нервного) импульса в ЦНС посредством афферентного нейрона;
- 2) второе звено — афферентный нейрон с отростками, его тело для соматической нервной системы расположено в спинномозговых ганглиях или ганглиях черепных нервов. Импульс поступает по дендриту афферентного нейрона, а затем по его аксону в ЦНС;
- 3) третье звено — вставочные нейроны ЦНС. Их назначение — обеспечение связи с другими отделами ЦНС, переработка и передача импульсов к эфферентному нейрону;
- 4) четвертое звено — эфферентный нейрон, он вместе с другими нейронами ЦНС перерабатывает информацию, формирует ответ в виде нервных импульсов, посылаемых к пятому звену рефлекторной дуги — эффектору (рабочему органу).

Различают огромное количество разнообразных рефлексов — от простых (двухнейронных, когда в рефлекторной дуге отсутствуют вставочные нейроны) до полисинаптических, проходящих через разные зоны коры головного мозга. По биологическому значению рефлексы делят на гомеостатические (регулирующие работу органов дыхания, сердца, секретиции и моторики пищеварительной системы), защитные (оборонительные), ориентировочные (непроизвольный поворот глаз и тела в сторону резкого раздражителя) и т. д.

Гуморальная регуляция осуществляется с помощью гормонов, тканевых гормонов, медиаторов и метаболитов. Часть медиаторов (ацетилхолин, катехоламины, серотонин), выделяемых нервными окончаниями в синаптическую щель, попадает в кровь, разносится по всему организму и действует на все органы и ткани, в которых имеются соответствующие рецепторы. Это действие происходит постоянно, оно дополняет действие нервной системы, реализуемой посредством нервно-мышечных синапсов и синапсов, образуемых на секреторных клетках (в основном, это пищеварительные железы). Таким образом, медиаторы, циркулирующие в крови, участвуют в поддержании исходного фона — рабочего состояния органов. Однако при увеличении активности нервной системы возрастает выброс медиаторов и количество их в крови, что также вносит некоторый вклад в изменение интенсивности работы органов [12].

Миогенный механизм регуляции бурно прогрессировал с развитием мышечной системы в процессе эволюции, что связано с интенсификацией двигательной активности животных. Организм человека примерно на 50 % состоит из мышц по своей массе. Это скелетная мускулатура (40 % массы тела), мышцы сердца и гладкие мышцы внутренних органов (сосудов, стенки желудочно-кишечного тракта, мочевого пузыря и др.). Сущность миогенного механизма регуляции состоит в том, что предварительное умеренное растяжение скелетной или сердечной мышц увеличивает силу их сокращений. Сократительная активность гладкой мышцы также зависит от степени ее растяжения. При увеличении наполнения органа тонус гладкой мышцы сначала возрастает, а затем возвращается к исходному уровню (пластичность гладкой мышцы), что обеспечивает регуляцию тонуса сосудов и наполнение внутренних полых органов без существенного повышения давления в них (до определенной величины). При уменьшении наполнения органа происходят противоположные изменения [6].

1.2. Высшая нервная и психическая деятельность

Высшая нервная и психическая деятельность человека формируется большими полушариями головного мозга и ядер ближайшей подкорки. Эта деятельность осуществляется совокупностью рефлексов

и высших психических функций, обеспечивающих индивидуальное приспособление к изменяющимся условиям среды, а также адекватное поведение во внешнем мире.

Все рефлексы подразделяют на две группы: врожденные (безусловные) и приобретенные (условные). *Условный рефлекс* — это приобретенная в онтогенезе реакция организма на раздражитель, ранее индифферентный для этой реакции. Руководствуясь таким подходом, различают низшую и высшую нервную деятельность. Под термином «высшая нервная деятельность» (ВНД) следует понимать совокупность нейрофизиологических процессов, обеспечивающих сознание, подсознательное усвоение информации и обучение в онтогенезе всем видам деятельности, в том числе и приспособительному поведению в окружающей среде. Психическая деятельность — это идеальная, субъективно осознаваемая деятельность организма, осуществляемая с помощью нейрофизиологических процессов ВНД [6].

Одна из моделей формирования поведения человека в результате ВНД была предложена П. К. Анохиным (рис. 1.2).

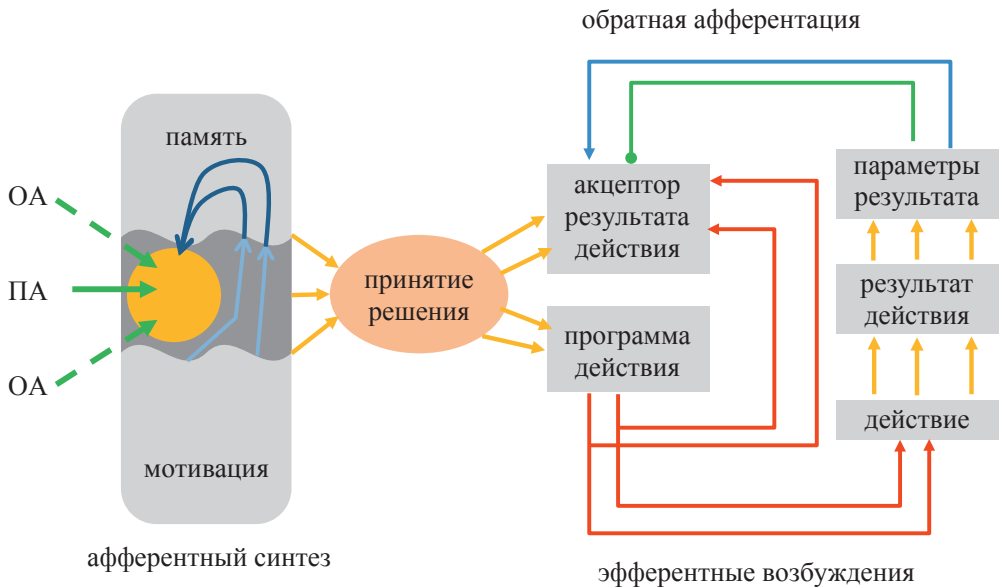


Рис. 1.2. Модель формирования поведенческого акта:
ОА — обстановочная афферентация; ПА — пусковая афферентация

Эта модель включает ряд важных понятий.

Афферентный синтез — переработка возбуждений, связанных с доминирующей мотивацией, обстановочной афферентацией, пусковым стимулом, взаимодействующими с аппаратом памяти. Под обстановочной афферентацией понимают совокупность всех внешних факторов и идущих от них воздействий. Комплекс обстановочной афферентации также активирует аппараты памяти о ранее совершаемых действиях в подобных ситуациях, что повышает шансы достичь полезного результата [8]. Возбуждение, возникающее в ответ на условный раздражитель, является пусковым, выводящим заранее сформированную доминирующую мотивацию на пороговый уровень.

Эфферентный синтез — включение процесса принятия решения и передача его к *акцептору результата действия* (АРД) и программе действия. Стадия принятия решения является результатом афферентного синтеза и обеспечивает формирование единственной совокупности эфферентных возбуждений, направляемых на достижение полезного результата. После этого все комбинации центральных возбуждений приобретают исполнительный характер. Вслед за принятием решения в функциональной системе поведенческого акта параллельно формируются два узловых механизма: аппарат АРД и программа действия.

АРД — это модель будущего результата и аппарат оценки физических, химических, биологических свойств полезного приспособительного поведенческого результата. В АРД происходит сличение обратной афферентации от параметров реальных достигнутых результатов с их предполагаемой нервной моделью [8]. Опыт человека убеждает в том, что, стремясь к достижению какой-либо цели, основные параметры цели конкретизируются изначально мысленно, т. е. идеально. В последующем нервная (идеальная) модель результата, сформированная в АРД, сопоставляется с реальным результатом. Если они не совпадают, то возникает ориентировочная реакция.

Программа действия представляет собой сопряжение и координацию нервных и гуморальных эфферентных механизмов управления работой периферических эффекторов в ходе реализации поведенческого акта для достижения полезного результата.

Поведенческий акт (действие) начинается после выхода программных возбуждений на периферические эффекторы (соматические, вегетативные) и обеспечивает получение полезного результата. Однако полученный результат может и не соответствовать запланированному.

Завершающей стадией является оценка параметров достигнутого результата деятельности в АРД. Это осуществляется с помощью сличения обратной афферентации о полученном результате с запрограммированным. Если полученный результат совпал с запланированным, АРД активизирует аппарат положительных эмоций; если полученный результат не совпал с запланированным, то возникает отрицательная эмоция, что побуждает организм к организации повторного, более совершенного поведенческого акта для достижения цели. Важно отметить, что в натуральных условиях жизнедеятельности человека при длительном или невозможном достижении необходимого результата эмоциональное напряжение может перерасти в эмоциональный стресс, который становится основой для возникновения и развития нарушений в работе целого организма, таких как артериальная гипертензия, язвенные поражения ЖКТ, иммунодефицитные состояния, тревожно-депрессивные расстройства.

В результате непрерывной работы ВНД в реальных условиях формируется *научение*, представляющее собой выработку приспособительных форм поведения организма, в том числе и навыков физического труда. При этом важное значение имеет формирование временной связи между центрами безусловного и условного раздражителей, процессы синаптической потенции и иррадиации возбуждения в коре головного мозга с образованием доминантного очага. Научение путем наблюдения и «правильного» действия организма в ситуации, с которой он впервые встретился, называется *когнитивным научением*. При когнитивном научении возможно два способа: *простое подражание* (дети подражают взрослым, получая навык владения речью), *викарное научение* (наблюдение за выполнением гимнастических упражнений и последующее их выполнение, если обучающийся их запомнил), *научение путем мышления* в ситуации, впервые встретившейся организму. Сложное приспособительное поведение строится из трех основных компонентов — инстинктов, условных рефлексов и мышления, благодаря которым сразу вырабатывается программа адаптивного поведения в новых условиях, внезапно сложившихся ситуациях и возможность выбора наиболее адекватной тактики поведения в новой обстановке с учетом прогноза развития событий.

Основой всех форм психической деятельности является сознание (идеальное субъективное отражение реальной действительности) [15], которое является продуктом ВНД. Сознание обуславливает способ-

ность мыслительного диалога с самим собой, т.е. ведет к появлению самосознания. ВНД протекает на уровне подсознания. Организм не осознает электрофизиологические, нейрохимические процессы, происходящие в ЦНС, а также структурные изменения, обеспечивающие долговременную память, но осознает ту или иную физическую и психическую деятельность. На уровне сознания протекают все формы психической деятельности человека: ощущение, внимание, мышление, восприятие, представление, эмоция и воля. Важно подчеркнуть, что ощущения возникают не только при действии внешних раздражителей, но при действии внутренних раздражителей, например, при повышении осмотического давления внутренней среды организма (жажда), при недостатке питательных веществ (чувство голода), при переполнении ряда внутренних органов (желудка, прямой кишки, мочевого пузыря), а также при патологии. Причем стимул должен иметь достаточную (пороговую) силу, информация от очень слабых раздражителей не осознается организмом, поскольку она не сопровождается возникновением ощущений. Неавтоматизированная физическая работа, требующая мышления и внимания, также осуществляется осознанно.

1.3. Системный принцип регуляции функций организма

Поддержание показателей внутренней среды организма человека осуществляется с помощью регуляции деятельности различных органов и физиологических систем, таких как сердечно-сосудистая система, система дыхания, лимфатическая система, метаболизм и терморегуляция, сенсорные системы, объединяемых в единую иерархически связанную функциональную систему.

Взаимосвязанная и нормальная жизнедеятельность всех составных частей организма возможна только при непременном условии сохранения относительного физико-химического постоянства его внутренней среды. Это постоянство имеет динамический характер, поскольку поддерживается не на абсолютно постоянном уровне, а в пределах допустимых колебаний основных физиологических функций. Такое состояние называется гомеостазом, который возможен благодаря механизмам регуляции и саморегуляции.

Единство регуляторных механизмов организма заключается в их взаимодействии. Так, в случае увеличения содержания углекислого газа в крови возбуждаются хеморецепторы аортальной и синокаротидной рефлексогенных зон, при этом увеличивается поток импульсов по соответствующим нервам в ЦНС, а оттуда — к дыхательной мускулатуре, что ведет к учащению и углублению дыхания. Углекислый газ действует на дыхательный центр и непосредственно, что тоже вызывает усиление дыхания. При действии холодного воздуха на терморецепторы кожи увеличивается поток афферентных импульсов в ЦНС, что ведет к выбросу гормонов, интенсифицирующих обмен веществ, и к увеличению теплопродукции. Ядра гипоталамуса вырабатывают нейrogормоны, регулирующие функцию эндокринных желез с помощью аденогипофиза [14].

Взаимосвязь различных систем, поддерживающих гомеостаз, иллюстрирует схема на рис. 1.3.

Системный принцип регуляции заключается в том, что различные показатели организма поддерживаются на оптимальном уровне с помощью многих органов и систем. Например, концентрация кислорода и углекислого газа в крови поддерживается за счет деятельности сердечно-сосудистой, дыхательной, нервно-мышечной системы, системой крови. Они объединяются в различные функциональные системы.



Рис. 1.3. Пример взаимосвязи физиологических систем, поддерживающих гомеостаз

Функциональные системы формируются из физиологических систем, которые являются генетически сформированной совокупностью

органов и тканей организма, выполняющей определенную функцию (например, система дыхания). Структура физиологической системы включает [12]:

- управляющее устройство — нервный центр, представляющий собой объединение ядер различных уровней ЦНС;
- выходные каналы (нервы и гормоны);
- исполнительные органы — эффекторы, обеспечивающие в ходе физиологической деятельности поддержание регулируемого процесса, показателя на некотором оптимальном уровне (полезный результат деятельности функциональной системы);
- рецепторы результата (сенсорные рецепторы) — датчики, воспринимающие информацию о параметрах отклонения регулируемого процесса, показателя от оптимального уровня;
- канал обратной связи (входные каналы), информирующий нервный центр с помощью афферентной импульсации от рецепторов результата о достаточности или недостаточности эффекторных усилий по поддержанию регулируемого процесса, показателя на оптимальном уровне. Эти импульсы по каналам обратной связи поступают в нервный центр, регулирующий тот или иной показатель.

Обратная связь осуществляется также с помощью гуморальных веществ. Метаболиты, попадая в кровь, а с кровью — в ЦНС, действуют и на соответствующие центры, изменение активности которых оказывает корригирующее влияние на органы и ткани организма. Метаболиты действуют также на сам работающий орган, что ведет к расширению сосудов и улучшению обмена веществ в органе, и на рецепторы рабочего органа (или органов) — рецепторы результата, что тоже отражается на активности рецепторов и, естественно, на импульсации в афферентных путях, проводящих импульсы в ЦНС по принципу обратной связи и способствующие нормализации регулируемого показателя.

Если же одной физиологической системы оказалось недостаточно и показатель организма остается повышенным или пониженным, то формируется функциональная система, включающая дополнительные физиологические системы — выделительную и (или) эндокринную.

Системообразующим фактором функциональной системы любого уровня является полезный для жизнедеятельности организма приспособительный результат, необходимый в данный момент, и доми-

нирующая мотивация. Все множество полезных приспособительных результатов можно объединить в две группы: поддержание постоянства внутренней среды организма и достижение результата в социальной деятельности.

Обеспечение жизнедеятельности организма в различных условиях среды обитания возможно только за счет запуска адаптивных механизмов регуляции функций организма. Состояние адаптивных механизмов — один из критериев здоровья человека. Адаптивные реакции обеспечивают гомеостазис, работоспособность, максимально возможную в конкретных условиях продолжительность жизни, репродуктивность. Адаптивные реакции возникают под влиянием различных социальных и природных факторов [15].

Природные факторы адаптации действуют на организм в совокупности или порознь: это фотопериодизм (день — ночь), продолжительность светового времени суток, зависящая от времени года и географического расположения места жительства, температура окружающей среды, действие воды и содержащихся в ней веществ, барометрическое давление, геомагнитные поля. Интенсивность воздействия на организм многих из этих факторов существенно зависит от гравитации и недостатка кислорода, космического излучения.

Социальные факторы адаптации связаны с трудовой деятельностью человека. К ним относятся: избыточный шум на рабочем месте, работа в горячих или холодных цехах, создающая дополнительную нагрузку на терморегуляторные механизмы; загрязнение среды обитания, что нередко ведет к различным заболеваниям, например, при вдыхании вместе с воздухом летучих токсичных веществ; работа под водой в условиях дыхания при повышенном давлении газовой смеси; гипокинезия и т. д.

Адаптивные реакции реализуются на клеточном, органном, системном и организменном уровнях. В зависимости от скорости включения и длительности действия врожденных и приобретенных механизмов адаптации следует различать три их группы: срочные, долговременные и постоянные.

Срочные адаптивные механизмы включаются сразу после начала действия безусловных и условных раздражителей. Например, с началом бега усиливается и учащается деятельность сердца, учащается дыхание. При действии на организм холода включаются терморегуляторные механизмы; при возникновении какой-либо опасности

человек стремится избежать ее, т. е. предотвратить влияние на организм неблагоприятного фактора — в этом случае поведенческая реакция имеет упреждающий характер. Если раздражающий фактор уже действует, то поведение направлено на избавление от него. Срочный адаптивный механизм может сработать заблаговременно по механизму условного рефлекса.

Проблема повышения устойчивости (резистентности) организма при действии различных факторов, стрессоров, является и в настоящее время актуальной. Стрессор — это любое сильное физическое или психологическое воздействие на организм, оказывающее часто отрицательное влияние — дистресс, реже положительное — эустресс (например, большая радость). При действии стрессора возникает состояние напряжения организма — стресс. Сильнодействующие стрессоры, оказывающие выраженное неблагоприятное влияние на организм, называют также экстремальными факторами, а ситуацию, когда они действуют, экстремальной. Стандартную реакцию на любой сильный раздражитель (холод, болевое раздражение, токсическое воздействие, сочетающееся с болью) обнаружил канадский ученый Ганс Селье: реакция включает увеличение коркового слоя надпочечников и повышение его активности; уменьшение вилочковой железы и лимфатических желез; точечные кровоизлияния и кровоточащие язвочки в слизистой оболочке желудка и кишечника. Однако патологических изменений в организме может и не наблюдаться, если стрессор недостаточно силен, а устойчивость (резистентность) организма высока. При длительном действии стрессора возникают адаптивные реакции организма, названные Г. Селье общим адаптационным синдромом. Он же впервые сформулировал концепцию стресса и показал, что болезнетворный фактор обладает пусковым действием, включающим выработанные в процессе эволюции механизмы адаптации [16].

Систематические физические упражнения делают организм более устойчивым к заболеваниям, повышают устойчивость организма к недостатку кислорода, улучшают утилизацию кислорода с помощью активизации ферментативных систем. Физические тренировки обеспечивают экономное расходование энергии, улучшают координационную деятельность ЦНС. Двигательная активность дает радость общения с природой, мышечную радость. Вследствие движений улучшается функциональное состояние ЦНС за счет проприоцептивной импульсации от мышц. Активируются и уравниваются возбуждательный

и тормозной процессы. В клетках коры большого мозга увеличивается содержание рибонуклеиновой кислоты, имеющей непосредственное отношение к механизмам памяти. Усиливаются ассоциативные процессы и творческая деятельность. В головном мозге увеличивается количество капилляров и их длина. Двигательная физическая активность уменьшает скорость склерозирования сосудов, поскольку при этом снижается уровень холестерина в крови. Своевременная активация мышечной деятельности постепенно нормализует функции всех систем организма и в сочетании с активной умственной работой обеспечивает гармоничное развитие личности.

Контрольные вопросы к главе 1

1. Дайте определение физиологической системе.
2. Перечислите основные принципы работы функциональных систем.
3. Опишите модель регуляции гомеостаза.
4. Перечислите основные звенья нервного механизма регуляции.
5. Оцените роль гуморальной и миогенной регуляции в обеспечении гомеостаза.
6. Перечислите разновидности рефлексов и их роль в организации жизнедеятельности человека.
7. Объясните понятие акцептора результата действия.
8. Перечислите адаптивные механизмы регуляции и опишите проблему стрессоустойчивости.

Глава 2.

Биомедицинские сигналы в задачах функциональной диагностики: принципы формирования исследований и их организация^{*}

Большинство медицинских исследований требует проведения длительного мониторинга состояния пациента, что связано с необходимостью применения автоматизированных систем регистрации биомедицинских сигналов (далее — биосигналы), их обработки для оценки функциональных изменений организма и анализа больших массивов данных для формирования стратегии лечебного процесса.

Биосигналы представляют собой *физические проявления физиологических процессов живого организма*, которые *могут быть измерены и представлены* в виде, удобном для последующей обработки. Обработка биосигналов проводится с целью выделения информативных признаков для медицинской диагностики.

По тому, каким образом в живом организме формируется биосигнал, можно выделить две основные группы биосигналов.

К **первой группе** относятся биосигналы, формируемые собственными физическими полями в тканях организма человека. Эти биосигналы параметрически модулируются физиологическими и функциональными процессами, формируемыми в организме. Их распределение в пространстве и изменение во времени несут важную биологическую информацию, которую можно использовать при научных исследованиях и в целях медицинской диагностики. Процедура измерения биосигна-

^{*} Авторы главы 2: В. С. Кубланов, А. Ю. Долганов (Уральский федеральный университет, г. Екатеринбург)

лов в этом случае может быть инвазивной или неинвазивной. Инвазивные процедуры предполагают имплантацию внутрь тела сенсорных устройств или других устройств, неинвазивные — используют сенсорные устройства, располагаемые на поверхности кожи, что минимизирует риск для пациента во время исследования.

Для формирования биосигналов **второй группы** необходимо воздействовать на биологические ткани внешними физическими полями.

Физиологические процессы в органах и тканях человека, вызванные электрохимическими и кинетическими процессами в них, являются относительно низкочастотными. При использовании неинвазивной процедуры регистрации биосигналов эндогенные электрические сигналы в значительной степени экранируются проводящими тканями биологического объекта с неоднородным распределением электрической проводимости. Но несмотря на это на кожном покрове формируются биоэлектрические потенциалы. Причем на определенных участках кожи формируется электрический потенциал, меняющийся с изменением электрофизических свойств определенного органа или системы при их функционировании.

Диапазон частот биосигналов, характеризующих функционирование органов и систем, лежит в полосе частот от долей Гц до единиц кГц. Разность квазистатических потенциалов между участками на кожном покрове человека достигает долей В и в значительной степени зависит от электродов, с помощью которых они регистрируются. Разность переменных потенциалов оценивается в диапазоне от единиц мкВ до десятков мВ.

При проведении регистрации на биосигнал всегда накладываются сигналы наводок (помех) и шумов. Наводки возникают вследствие действия внешних физических полей, не имеющих прямого отношения к объекту исследований. Сигналы наводок (помех) и шумов оказывают влияние на чувствительный элемент измерительного преобразователя, на отдельные узлы или цепи устройства преобразования биосигнала. Под шумами понимаются такие сигналы, которые появляются на выходе вследствие особенностей функционирования и параметров измерительной аппаратуры, а также вследствие работы других подсистем и наличия процессов в организме, в результате которых возникают сигналы, не имеющие прямого отношения к определяемым показателям или характеристикам. Так, например, если при измерении малых разностей потенциалов между участками кожного по-

крова электроды будут непрерывно колебаться из-за колебаний кожи, то при больших переходных сопротивлениях в месте контакта электродов с кожей и при нестабильности контактных явлений аппаратура покажет наличие переменного сигнала, появившегося в результате взаимодействия чувствительного элемента (электрода) с объектом измерений и не характерного для объекта, находящегося в нормальном состоянии.

В медицинской практике шумы биологического происхождения, вызванные процессами, не имеющими прямого отношения к определяемым параметрам или характеристикам, называют часто артефактами. К артефактам биологического происхождения, как правило, относятся помехи, обусловленные дыханием или движениями обследуемого во время регистрации биосигналов, а также любая активность систем организма, не связанная с регистрируемым процессом, но оказывающая влияние на определяемые значения диагностических показателей. Наиболее ярким примером таких процессов может служить миографическая активность периферических мышц при регистрации электрокардиографического сигнала.

Очень часто трудно отличить присутствующие помехи и шумы от биосигналов, появившихся вследствие взаимодействия с объектом измерения чувствительного элемента измерительного преобразователя. Поэтому, даже располагая аппаратурой с гарантированными метрологическими характеристиками, нельзя с полной уверенностью утверждать, что погрешность результатов измерений не превышает значений, нормированных для технического измерительного средства.

Еще одним фактором, важным при исследовании биологических организмов, является их изменчивость и индивидуальность параметров и показателей. Даже на групповом уровне проявляется зависимость от национальных, возрастных, генетических и климатических особенностей, поэтому корректным является описание свойств биосигналов у группы организмов, в которой проводятся исследования одних и тех же параметров.

Для установления каких-либо закономерностей в медицинской диагностике широко применяются специализированные методы обработки биосигналов. Это обусловлено тем, что из-за субъективности и многофакторности получаемых результатов установить объективные закономерности можно только после математической обработки достаточно большого массива информационного материала. Получе-

ние такого фактического материала часто затруднительно. И даже в тех случаях, когда определение интересующего параметра или показателя можно выполнить относительно быстро, набор информационного материала, анализ полученных данных с целью установления объективных закономерностей занимают значительные промежутки времени.

Наибольшую диагностическую ценность имеют переменные биосигналы, характеризующие функционирование сердца, центральной нервной системы, опорно-двигательного аппарата, состояние нервно-мышечной проводимости, вестибулярного аппарата и др.

Биосигналы функциональной диагностики первой группы:

- электрокардиограмма;
- электроэнцефалограмма;
- сигнал кожно-гальванической реакции;
- стабิโลграфический сигнал;
- речевой сигнал.

Биосигналы функциональной диагностики второй группы:

- реографический сигнал;
- фотоплетизмографический сигнал;
- лазерная доплеровская флоуметрия;
- пульсоксиметрия;
- ультразвуковая доплерография сосудов головного мозга и шеи.

Рассмотрим краткие характеристики некоторых из методов формирования биосигналов первой и второй групп, регистрация которых обеспечивается приборами функциональной диагностики Научно-исследовательского медико-биологического инженерного центра высоких технологий Уральского федерального университета.

2.1. Электрокардиограмма

Электрокардиограмма (ЭКГ) является электрическим проявлением сократительной активности сердца и может быть достаточно легко записана с помощью поверхностных электродов, помещенных на конечности или грудь. Форма волн ЭКГ изменяется под действием сердечнососудистых заболеваний и патологий, таких как ишемия миокарда и инфаркт, гипертрофия желудочков, а также нарушения проводимости.

Сердце представляет собой четырехкамерный насос с двумя предсердиями для сбора крови и двумя желудочками для выталкивания крови. Фаза отдыха или наполнения сердечной камеры называется диастолой, фаза сокращения — систолой. Правое предсердие (ПП, или *atrium*) собирает загрязненную кровь из главной и поллой вены (*vena cavae*). За время сокращения предсердия кровь проходит через правое предсердие к правому желудочку (*ventrium*) через трехстворчатые клапаны. В период желудочковой систолы загрязненная кровь выталкивается из правого желудочка (ПЖ) через легочный клапан к легким для очистки (оксигенации). Левое предсердие (ЛП) принимает очищенную кровь из легких, проходящую при сокращении предсердий к левому желудочку (ЛЖ) через митральный клапан. Левый желудочек является самой большой и наиболее важной камерой сердца. Сокращение левого желудочка является наиболее сильным по сравнению со всеми другими камерами сердца, так как он должен прокачивать насыщенную кислородом кровь через клапан аорты в аорту, преодолевая давление всей остальной сосудистой системы тела. В силу того, что уровень важности сокращений желудочков самый высокий, термины *систола* и *диастола* по умолчанию используются по отношению к желудочкам.

Сердечный ритм, или частота сердечных сокращений (ЧСС), управляется специальными клетками водителя ритма, которые образуют синоатриальный (СА) узел, расположенный в месте соединения главной вены и правого предсердия [29]. Частота срабатывания СА-узла управляется импульсами от вегетативной и центральной нервной системы, приводя к выработке нейротрансмиттерами ацетилхолина (для вагусной стимуляции, вызывающей снижение сердечного ритма) или адреналина (для симпатической стимуляции, вызывающей повышение сердечного ритма). Нормальная ЧСС в покое составляет около 70 уд./мин. Слишком низкая ЧСС, менее 60 уд./мин в период активности, может означать нарушение, называемое брадикардией. В ходе интенсивных упражнений или занятий спортом мгновенное значение ЧСС может достигать 200 уд./мин. Высокая ЧСС в покое может быть вызвана заболеванием или сердечной аномалией, которая называется тахикардией.

Координированные электрические процессы и специализированная проводящая система, свойственная только сердцу, играют главную роль в ритмической сократительной активности сердца. СА-узел является основным естественным водителем ритма, который выраба-

тывает собственную последовательность потенциалов действия. Потенциал действия СА-узла распространяется по всему объему сердца, вызывая особый вид возбуждения и сокращения.

В сердечном цикле в норме наблюдается следующая последовательность событий и волн (рис. 2.1).

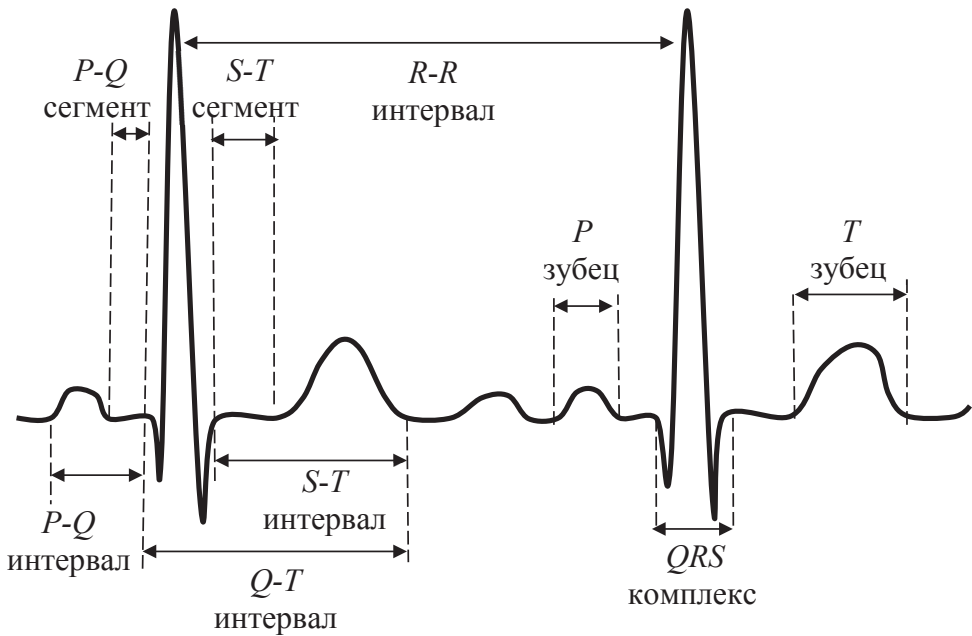


Рис. 2.1. Электрокардиограмма

Электрическая активность распространяется по мускулатуре предсердия со сравнительно небольшой скоростью, вызывая медленно продвигающуюся деполяризацию (сокращение предсердия). Это приводит к появлению Р-зубца в кардиограмме. Ввиду медленно-го сокращения предсердий и их небольшого размера Р-зубец представляет собой низкоамплитудную волну 0,1–0,2 мВ длительностью примерно 60–80 мс.

Волна возбуждения наталкивается на задержку распространения в артериовентрикулярном (АВ) узле, что, как правило, проявляется в ЭКГ в виде изоэлектрического сегмента продолжительностью 60–80 мс, следующего после Р-зубца и известного как PQ-сегмент. Эта пауза помогает завершению перемещения крови от предсердий к желудочкам и оканчивается возбуждением АВ-узла.

Пучок Гиса, ножки пучка Гиса и система специализированных проводящих волокон Пуркинье с большой скоростью распространяют стимул по желудочкам. Затем стимулирующая волна распространяется от верхушки сердца вверх, вызывая быструю деполяризацию (сокращение желудочков). На ЭКГ это проявляется в виде *QRS*-комплекса — острой двухфазной или трехфазной волны с амплитудой около 1 мВ и длительностью 80 мс.

Для мышечных клеток желудочков характерна относительно большая длительность потенциала действия — 300–350 мс. Плато на потенциале действия вызывает обычно изоэлектрический сегмент длительностью 100–120 мс, следующий после *QRS*-комплекса и известный как *ST*-сегмент.

Реполяризация (расслабление) желудочков проявляется в виде медленного *T*-зубца с амплитудой 0,1–0,3 мВ и длительностью 120–160 мс.

Любые нарушения в регулярной ритмической активности сердца называются аритмиями. Сердечная аритмия может быть вызвана нерегулярным срабатыванием СА-узла или аномальной и дополнительной стимулирующими активностями других отделов сердца.

Для измерения разности потенциалов на различные участки тела накладываются электроды. Так как плохой электрический контакт между кожей и электродами создает помехи, то для обеспечения проводимости на участки кожи в местах контакта наносят токопроводящий гель.

При функциональных исследованиях представляют интерес *исследования изменений некоторых параметров кардиоимпульсов*, таких как *QRS*-комплекс, *ST*-сегмент, *PQ*-сегмент, интервалы *PQ*, *ST*, *QT*, зубцы *P*, *Q*, *R*, *S*, *T*, *U* и вариабельность сердечного ритма (*R-R* интервалы).

При обработке ЭКГ первоочередной задачей является обнаружение *QRS*-комплексов и разметка их временного положения (формирование ритмограммы). Подробный обзор работ в этой области содержится в [17]. Развитие этих алгоритмов отражает огромные достижения в области компьютерных технологий. Известно много подходов для обнаружения *QRS*-комплексов, например, алгоритмы, использующие искусственные нейронные сети [18], генетические алгоритмы [19], вейвлет-преобразования [20], банки фильтров [21], а также эвристические методы, основанные главным образом на нелинейных преобразованиях элементов [22–24]. В работе [25] проведен сравнительный анализ упомянутых выше алгоритмов и алгоритмов, основанных на использовании производной обрабатываемого сигнала.

Анализ длительных записей ЭКГ показывает, что в них можно выделить участки со сравнительно медленными изменениями формы электрокардиосигнала во времени и интервалы, на которых происходит резкое изменение формы, связанное с патологическими явлениями. В первой ситуации можно использовать алгоритмы, рассчитанные на знание формы обнаруживаемого сигнала. Для второй ситуации — алгоритмы, слабо чувствительные к форме сигнала и реагирующие на признаки почти всегда присущие *QRS*-комплексу: высокий уровень по сравнению с другими элементами электрокардиосигнала и высокая скорость изменения (значение модуля производной). К таким алгоритмам относится алгоритм Пана-Томпкинса [26].

Сигнал variability сердечного ритма (BCP), определяемый изменениями межимпульсных *R-R* интервалов ЭКГ, является одним из наиболее важных маркеров состояния вегетативной нервной системы. В настоящее время из всей ЭКГ чаще всего берется на исследование *R-R* интервал. Используемые сегодня методики подразумевают измерение общей variability *R-R* интервалов и амплитуды формирующих ее колебательных составляющих. Обнаружение *R*-зубцов ЭКГ и измерения продолжительностей кардиоинтервалов в современных диагностических комплексах осуществляются программными способами с погрешностью не более ± 1 мс [26].

Детальный анализ сигналов BCP позволил разработать подход, основанный на положениях биологической кибернетики. Наиболее простой является двухконтурная модель регуляции сердечного ритма. Два контура (центральный и автономный) взаимосвязаны и регулируют синусовый узел. Симпатические нервные и гуморальные связи обеспечивают прямую связь между центральным и автономным контурами. При этом обратная связь осуществляется путем афферентных импульсов, приходящих от рецепторных зон органов и тканей, хеморецепторов и барорецепторов [27].

BCP отражает работу «сердечно-сосудистой системы и работу механизмов регуляции целостного организма, в частности общей активности регуляторных механизмов, нейро-гуморальной регуляции сердца, соотношения между симпатическим и парасимпатическим отделами вегетативной нервной системы» [28]. Нарушение этих систем регуляции происходит при психоэмоциональном стрессе и психоневрологических заболеваниях, заболеваниях сердечно-сосудистой системы, а также нарушениях, которые могут опосредованно влиять на состо-

яние вегетативной и центральной нервной системы и на механизмы гуморальной регуляции.

На протяжении многих лет ВСП является одним из самых многообещающих маркеров функционирования сердечной деятельности. Неинвазивность, простота в получении сигналов, чувствительность делают вариабельность сердечного ритма практичным и широко применимым методом. В некоторых работах, посвященных оценке ВСП, указано на значимость этого сигнала в получении знаний об изменениях в механизмах контроля артериального давления при гипертонии [29]. Параметры ВСП могут быть чувствительными к нарушениям регуляции вегетативной нервной системы и выступать в качестве ранних индикаторов этих нарушений [30].

Более подробно параметры ВСП рассмотрены в главе 4.

2.2. Электроэнцефалограмма

Электроэнцефалограмма (ЭЭГ) представляет собой запись электрической активности мозга. Организация мозга имеет несколько следующих важных аспектов. Главными частями мозга являются: кора головного мозга, мозжечок, ствол мозга (включающий средний мозг, продолговатый мозг и ретикулярное образование) и таламус (между средним мозгом и полусферами). Сигнал ЭЭГ может быть использован для изучения нервной системы, мониторингового наблюдения стадий сна, биологической обратной связи и управления, а также диагностики таких заболеваний, как эпилепсия.

Регистрация ЭЭГ производится электроэнцефалографом через специальные электроды. В настоящее время чаще всего используется расположение электродов по международным системам «10–20 %» или «10–10 %». Каждый электрод подключен к усилителю. Запись потенциалов с каждого электрода осуществляется относительно нулевого потенциала референта, за который, как правило, принимается мочка уха или сосцевидный отросток височной кости (*mastoid*), расположенный позади уха и содержащий заполненные воздухом костные полости.

Наименование «10–20» отражает то обстоятельство, что электроды располагаются вдоль центральной линии со значениями шага, равными 10, 20, 20, 20, 20 и 10 % от общего расстояния между носом и затыл-

ком; остальные цепочки электродов также располагаются в позициях, соответствующих аналогичным долям расстояния. Межэлектродные расстояния равны между собой как вдоль переднезадней линии, так и вдоль поперечной линии. Кроме того, позиции электродов симметричны.

Типичная комплектация оборудования для ЭЭГ включает низкочастотный фильтр с частотой среза 75 Гц. Длительность записей обычно составляет 10–20 мин одновременно по 8–16 каналам. Наблюдение стадий сна и выявление преходящих событий, связанных с эпилептическими припадками, может потребовать многоканальной записи ЭЭГ на протяжении нескольких часов.

Специализированные методики съема ЭЭГ предполагают использование игольчатых электродов, носоглоточных электродов, записи электрокортикограммы (ЭКоГ) от вскрытой части коры мозга и использование интрацеребральных электродов. Методика регистрации отклика на вызванные события включает первоначальную запись в покое (глаза открыты, глаза закрыты), гипервентиляцию (после эпизода глубокого дыхания с частотой 20 вдохов-выдохов в мин на протяжении 2–4 мин), фотостимуляцию (1–50 вспышек света в с), аудиостимуляцию громкими щелчками, сон (различные стадии) и фармакологические (лекарственные) пробы.

В сигналах ЭЭГ могут наблюдаться несколько типов ритмической или периодической активности (рис. 2.2).

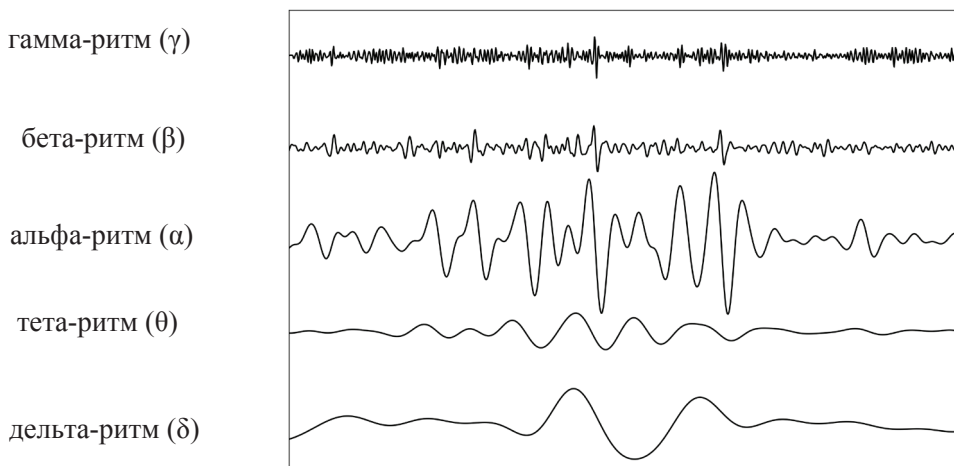


Рис. 2.2. Ритмы ЭЭГ практически здорового испытуемого В., 25 лет, мужчина

При анализе ЭЭГ обычно используются следующие диапазоны частот:

- 1) $0,5 < f < 4$ Гц — дельта-ритм (δ): состоит из волн амплитудой до сотни мкВ. Возникает как при глубоком естественном сне, так и при наркотическом, а также при коме. Дельта-ритм также наблюдается при регистрации ЭЭГ от участков коры, граничащих с областью травматического очага или опухоли;
- 2) $4 < f < 8$ Гц — тета-ритм (θ): высокий электрический потенциал — 100–150 мкВ, высокая амплитуда волн. Наиболее ярко тета-ритм выражен у детей (от 2 до 8 лет);
- 3) $8 < f < 13$ Гц — альфа-ритм (α): средняя амплитуда — 30–70 мкВ, могут однако наблюдаться высоко- и низкоамплитудные α -волны. Регистрируется у 85–95 % здоровых взрослых. Лучше всего он выражен в затылочных отделах;
- 4) $13 < f < 30$ Гц — бета-ритм (β): в состоянии активного бодрствования напряжение — 5–30 мкВ. Наиболее сильно этот ритм выражен в лобных областях, но при различных видах интенсивной деятельности резко усиливается и распространяется на другие области мозга. Выраженность β -ритма возрастает при предъявлении нового неожиданного стимула, в ситуации внимания, при умственном напряжении, эмоциональном возбуждении. Бета-волны по форме близки к треугольным вследствие заостренности вершин;
- 5) $30 \text{ Гц} < f < 120\text{--}170$ Гц — гамма ритм (γ): по данным некоторых авторов, частота гамма-ритма может достигать 500 Гц, амплитуда очень низка, ниже 10 мкВ, и обратно пропорциональна частоте. В случае если амплитуда гамма-ритма выше 15 мкВ, то ЭЭГ рассматривается как патологическая. Гамма-ритм наблюдается при решении задач, требующих максимального сосредоточенного внимания. Существуют теории, связывающие этот ритм с работой сознания.

2.3. Сигналы кожно-гальванической реакции

Кожно-гальваническая реакция (КГР), или вызванные кожные вегетативные потенциалы, характеризует электрические свойства кожи, связанные, главным образом, с активностью потовых желез, которые, в свою очередь, находятся под контролем симпатической нервной си-

стемы. Плотность потовых желез на поверхности тела неравномерна — на ладонях и подошвах около 400 на 1 см² поверхности кожи, на лбу — около 200, на спине — около 60. Существуют два типа потовых желез: апокринные и экринные.

Первые, расположенные в подмышечных впадинах и паху, определяют запах тела и реагируют на раздражители, вызывающие стресс. Они непосредственно не связаны с регуляцией температуры тела. Вторые расположены по всей поверхности тела, и их главная функция — терморегуляция. Однако те экринные железы, которые расположены на ладонях и подошвах ног, а также на лбу, реагируют в основном на внешние раздражители и стрессовые воздействия. В психофизиологии электрическую активность кожи используют как показатель ориентировочной или оборонительной реакции на внешние стимулы и показатель внутреннего эмоционального напряжения.

В качестве стимулов обычно используется раздражение кожи электрическим током, вспышка света, звуковой сигнал или глубокий вдох. Однако очень часто наблюдаются выраженные изменения электрической активности кожи, не вызванные каким-либо внешним раздражителем. Такие проявления, в отличие от вызванной реакции, называют *спонтанной активностью*, которая имеет эмоциональную и терморегуляционную компоненты. Эмоциональная активность регистрируется на ладонях и подошвах в обычных температурных условиях и усиливается в ответ на внешние возбуждающие стимулы. Терморегуляторная компонента появляется исключительно при возбуждении температуры окружающей среды.

Кожногальваническую реакцию связывают с секреторной деятельностью потовых желез, расположенных под электродами и контролируемые непосредственно ЦНС. Резкое падение сопротивления кожи является показателем эмоциональной активации в момент принятия решения. Диапазон изменения амплитуды сигнала кожногальванической реакции составляет 0,1–2 мВ, а частотный диапазон — от 0,1 до 10 Гц. Регистрация и обработка сигнала кожногальванической реакции используется в диагностике психоэмоционального состояния человека. В общем случае КГР может рассматриваться как показатель неспецифической нервно-психической напряженности и эмоциональности.

Чтобы изучить электродермальную активность, сигнал делится на уровень проводимости кожи, *skin conductivity level (SCL)* и реакцию проводимости кожи, *skin conductivity response (SCR)*. Они обычно упо-

минаются как тонические и фазные компоненты электродермальной активности [31]. Тоническая активность — это медленно меняющийся базовый сигнал, не связанный напрямую со стимулом. Частотные составляющие *SCLE* ниже 0,02 Гц. Фазовый компонент — это результат активации симпатической системы, более быстрый сигнал ($f < 0,5$ Гц) по сравнению с сигналом тонического компонента. Этапная активация может происходить после предъявления стимула (экзогенная активация) или самопроизвольно через нормальную регуляторную функцию симпатической системы (эндогенная активация).

Типичная структура сигнала реакции проводимости кожи представляет собой небольшой выступ, который перекрывает уровень проводимости кожи. *SCR* состоит из двух зон: зоны подъема и зоны распада. На рис. 2.3, *а* представлен пример сигнала КГР, на рис. 2.3, *б* показаны *SCR*, выделенные алгоритмическим способом.

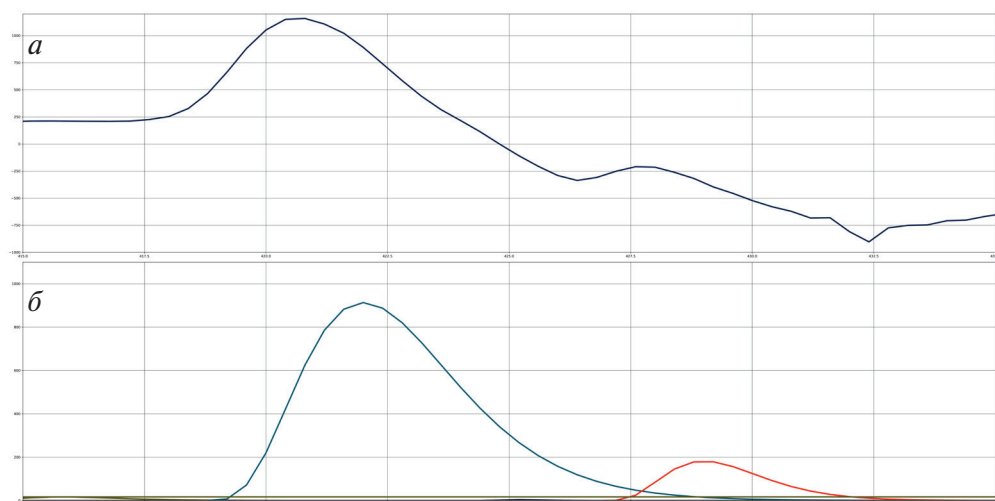


Рис. 2.3. Электрическая активность кожи:
а — сигнал КГР; б — компонента *SCR*

2.4. Стабилографический сигнал

Метод основан на регистрации параметров перемещения центра давления стоп пациента на плоскость стабилметрической платформы в процессе поддержания равновесия тела с последующей обра-

боткой полученных данных и позволяет выявлять наиболее ранние нарушения в позвоночнике, стопах, зрительном анализаторе и глазодвигательных мышцах, вестибулярном аппарате, зубочелюстной системе и даже в центральной нервной системе. Повреждения на любом из этих уровней проявляются нарушением равновесия, порой незаметным для самого обследуемого. Это уникальный метод, который позволяет зафиксировать и количественно определить имеющиеся нарушения равновесия, выявить причину неустойчивости.

Удержание человеком вертикальной позы сопровождается его микроколебательным процессом, зачастую малозаметным при визуальном наблюдении. Процесс отклонения тела человека от вертикали является абсолютно необходимым для восстановления утрачиваемого равновесия и обеспечения устойчивости [32]. Наиболее точным методом, позволяющим оценить запас устойчивости, исследовать качество функции равновесия, изучить вклад различных сенсорных систем в поддержание вестибулярной стойки, признано стабилметрическое исследование [33].

Стабилметрическое исследование — это метод исследования функций организма, связанных с поддержанием равновесия. Во время этого исследования оценивается поведение вестибулярной функции человека путем регистрации отклонений центра давления (ЦД). Во время исследования формируется стабiloграфический сигнал, который описывает положение ЦД во времени в декартовой системе координат или, как принято в биомеханике, во фронтальном (ось X) или сагиттальном (ось Y) направлениях. На рис. 2.4 изображена статокинезиограмма в декартовой системе координат, измеряющаяся в мм. По горизонтали — ось X , по вертикали — ось Y [34].

На рис. 2.5 представлены графики компонент стабiloграфического сигнала, где компонента X соответствует стабiloграфическому сигналу по оси X , а компонента Y соответствует стабiloграфическому сигналу по оси Y .

Стабилметрическое исследование является характеристикой баланса тела и применяется для выявления:

- функциональных нарушений при заболеваниях позвоночника, нервной системы, вестибулярного и зрительного анализатора;
- опороспособности конечностей;
- функционального состояния голеностопных суставов.

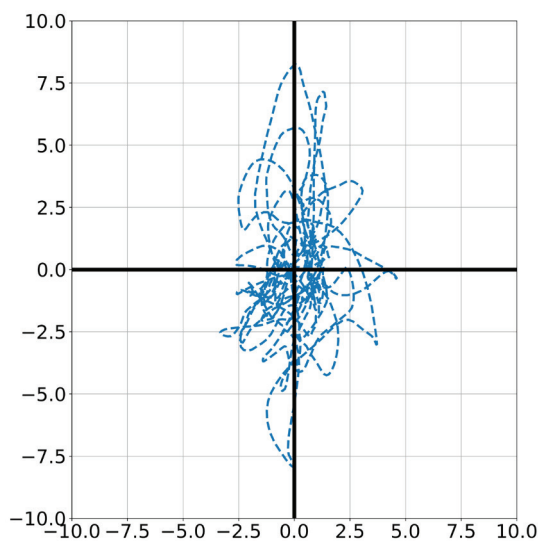


Рис. 2.4. Пример статокинезиограммы

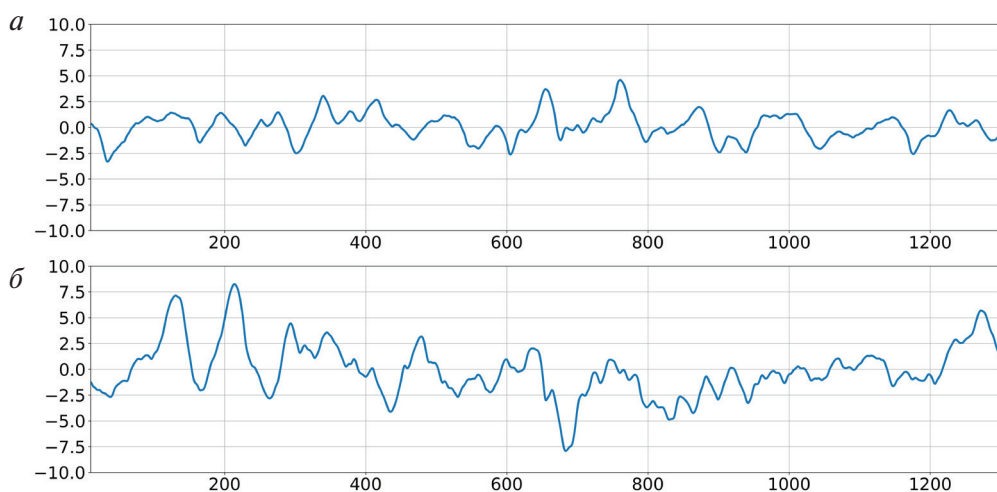


Рис. 2.5. Графики компонент стабиграфического сигнала:

a — компонента X ; b — компонента Y .

Стабиграфические параметры равновесия являются интегральной реакцией статокоординаторной функциональной системы, конечный полезный результат действия которой заключается в обеспечении равновесия и координации движений в процессе жизнедеятельности человека [35].

К достоинствам компьютерной стабิโลграфии следует отнести [36]:

- комфортность проведения исследования;
- малое время исследования;
- информативность исследования;
- многофункциональность при использовании функционально-нагрузочных проб.

В настоящее время компьютерная стабิโลграфия применяется при [36]:

- диагностике нарушений функции равновесия тела человека;
- реабилитации функции равновесия после травм и заболеваний и оценке результативности лечения;
- профессиональном отборе;
- экспертизе трудоспособности и профориентации;
- оценке качества функции равновесия и прогнозировании профессионального роста спортсменов, артистов балета;
- улучшении координации движений спортсменов;
- фундаментальных научных исследованиях биомеханики движений в аэрокосмической медицине, психофизиологии, валеологии и др.

Значимый вклад в результативность спортивной деятельности вносит система поддержания равновесия тела путем динамической стабилизации его положения относительно вектора гравитации [37].

Состояние статокINETической устойчивости спортсменов является предметом значительного количества исследований [38–40] при оценке уровня переносимости тренировочных и соревновательных нагрузок, выборе спортивной специализации в видах спорта, где умение сохранять равновесие определяет итоговый результат.

Контроль функционального состояния спортсменов на основе методов и средств компьютерной стабิโลграфии обладает высокой степенью чувствительности к отклонениям функционального состояния, возможностью формирования индивидуальных и групповых нормативов, а также способен отслеживать текущее состояние спортсмена [41].

2.5. Речевой сигнал

Речевой сигнал может использоваться как диагностический в случаях, когда необходимо исследовать нарушение звукового и речевого трактов, а также изменения функционального состояния, например, при депрессивно-тревожных расстройствах.

Звуки речи образуются за счет прохождения выдыхаемого воздуха из легких в голосовой тракт (а также для некоторых звуков — через носовой тракт). Голосовой тракт начинается голосовой щелью в горле и заканчивается губами и ноздрями. Форма голосового тракта меняется, производя различные типы звуковых элементов, или фонем, которые, складываясь, формируют речь. Фактически голосовой тракт действует как фильтр, который модулирует спектральные характеристики входящих потоков воздуха. Очевидно, что эта система является динамической и что данный фильтр и, следовательно, производимый им сигнал имеют изменяющиеся во времени характеристики, т. е. они являются нестационарными.

Звуки речи в самом общем виде могут быть классифицированы как гласные, фрикативные и взрывные. В формировании гласных звуков участвует голосовая щель: воздух проталкивается через голосовые нити, которые поддерживаются в состоянии некоторого напряжения. Результатом является серия квазипериодических импульсов воздуха, которые проходят через голосовой тракт. Входной поток голосового тракта может рассматриваться как последовательность импульсов, которые являются почти периодическими. За счет свертки с импульсной характеристикой голосового тракта, который постоянно поддерживается в определенной конфигурации на протяжении всей длительности гласного звука, формируется некоторый квазипериодический сигнал с характерными повторяющимися формами волн. На рис. 2.6 показана запись произнесенной фразы.

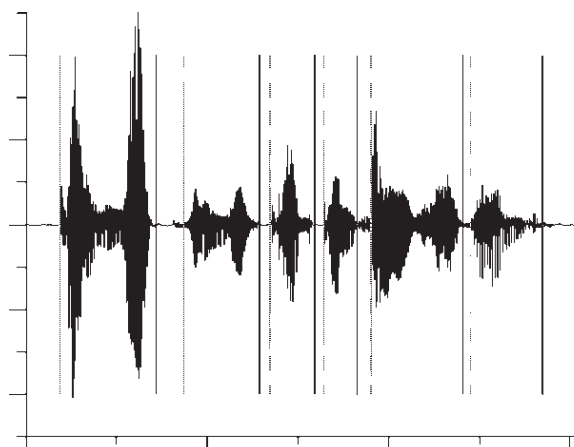


Рис. 2.6. Речевой сигнал

Исследуемыми характеристиками гласного сигнала являются тон (средний интервал повторения импульсной характеристики звукового тракта или базовой волновой формы) и резонансные или формантные частоты системы голосового тракта [42].

2.6. Биосигналы функциональной диагностики второй группы

Реографический сигнал

Представляет собой изменение во времени электрического сопротивления участка биологической ткани, расположенного между измерительными электродами. Для регистрации реографического сигнала через участок исследуемых биологических тканей пропускается переменный электрический ток с частотой порядка нескольких сотен кГц и амплитудой не превышающей 1 мА. Амплитуда реографического сигнала измеряется как падение напряжения на участке биологических тканей, расположенных между измерительными электродами. Частотный диапазон этого биосигнала составляет 0,3...70 Гц. Методы реографии используются в кардиологической практике для определения параметров центрального кровотока (по Тищенко), например величины сердечного выброса с помощью дифференциальной реограммы, и параметров периферического кровотока, например, формы пульсовой волны, величины индекса перфузии.

Фотоплетизмографический сигнал

Представляет собой изменение во времени объема кровеносного сосуда под действием пульсовых волн. Для регистрации фотоплетизмографического сигнала через исследуемый участок биологических тканей пропускается поток излучения оптического или инфракрасного диапазона. Величина сигнала измеряется как ослабление излучения, проходящего через исследуемый участок биологической ткани, содержащей кровеносный сосуд (или отраженного от участка исследуемой биологической ткани). Амплитуда сигнала при использовании широкополосного фотоприемника составляет не менее 0,1 мВ. Частотный диапазон составляет 0,3...70 Гц. Методы фотоплетизмографии исполь-

зуются в кардиологической практике для определения параметров периферического кровотока, например, с целью определения эластических свойств сосудов. В клиническом мониторинге используется при построении пульсоксиметров для неинвазивного мониторинга степени насыщения крови кислородом.

Лазерная доплеровская флоуметрия

Метод лазерной доплеровской флоуметрии основывается на оптическом неинвазивном зондировании тканей лазерным излучением и анализе рассеянного и отраженного от движущихся в тканях эритроцитов излучения. Отраженное от статических (неподвижных) компонентов ткани лазерное излучение не изменяет своей частоты, а отраженное от подвижных частиц (эритроцитов) — имеет доплеровское смещение частоты относительно зондирующего сигнала. Переменная составляющая отраженного сигнала определяется двумя факторами: концентрацией эритроцитов в зондируемом объеме и их скоростью. Глубина оптического зондирования ткани зависит от длины волны лазерного источника и от типа ткани. Для лазерного излучения на длине волны 632 нм она составляет около 1 мм.

В анализаторе «ЛАЗМА ПФ» реализован метод диагностики с одновременной регистрацией температуры области исследований. Прибор работает без оптического волокна, также применяется беспроводная передача информации в компьютер, планшет или смартфон. Анализатор может широко применяться как в стационарах и поликлиниках, так и в домашних условиях.

С помощью портативных анализаторов можно организовать распределенную систему анализаторов для одновременного контроля микроциркуляции в нескольких областях (до четырех), например, на верхних и нижних конечностях. Такой подход позволяет исследовать системные особенности микроциркуляции.

Пульсоксиметрия

Основу метода пульсоксиметрии составляет измерение поглощения света определенной длины волны гемоглобином крови. Гемоглобин служит своего рода фильтром, причем «цвет» и «толщина» этого естественного фильтра могут меняться. «Цвет» фильтра зависит от ко-

личества кислорода, связанного с гемоглобином, или, иными словами, от процентного содержания оксигемоглобина. На этом базируется способность пульсоксиметра устанавливать степень оксигенации крови. На изменение «толщины» фильтра влияет пульсация артериол: каждая пульсовая волна увеличивает количество крови в артериях и артериолах, что позволяет производить оценку частоты пульса и амплитуды пульсовой волны.

Таким образом, пульсооксиметр позволяет определить сразу три диагностических параметра: степень насыщения гемоглобина крови кислородом, частоту пульса и его «объемную» амплитуду.

Ультразвуковая доплерография сосудов головного мозга и шеи

Допплерография сосудов головного мозга, шеи, верхних и нижних конечностей, как и их дуплексное сканирование, относится к неинвазивным диагностическим процедурам. Их преимуществом являются доступная стоимость и отсутствие противопоказаний, высокая информативность.

Сущность доплеровского эффекта состоит в том, что от движущихся объектов ультразвуковые волны отражаются с измененной частотой. Этот сдвиг частоты пропорционален скорости движения лоцируемых структур: если движение направлено в сторону датчика, то частота увеличивается, если от датчика — уменьшается. Использование эффекта Доплера позволяет вычислить скорость кровотока, определить его нарушение в отдельных сосудах. Чаще всего этих данных достаточно, чтобы врач поставил точный диагноз. В свою очередь дуплексное сканирование сосудов шеи, головы и конечностей дает информацию не только о качестве кровотока, но и о геометрии сосудистого просвета, извилистости русла, наличии анатомических или послеоперационных аномалий, толщине стенок, появлении тромбов и атеросклеротических бляшек.

2.7. Организация функциональных исследований

Мониторинг и регистрация биосигналов являются естественным продолжением медицинских исследований с учетом постоянного развития технологий.

Обычная практика для медицинских тестов — это исследование в определенный момент времени, когда врач осматривает пациента. Помимо клинического опроса врач проверяет пульс, измеряет артериальное давление, берет образец крови и иногда мочи, а также, возможно, измеряет температуру тела и анализирует потоотделение. Эта собранная информация используется для постановки диагноза или, если этого недостаточно, для запроса дополнительных исследований.

В качестве дополнительных могут быть функциональные исследования, например, электрокардиография, электроэнцефалография, фотоплетизмография, проверка функции легких или функциональные пробы с дозированной физической нагрузкой, а также исследования, нацеленные на получение изображений, такие как рентгенологические, ультразвуковые, эндоскопические или ангиографические.

Все эти исследования являются, по сути, точечными измерениями, даже если они включают генерацию изображения или функциональные исследования в течение короткого периода времени. Эти точечные меры используются для постановки медицинского диагноза. Основываясь на диагнозе, врач пытается предсказать изменения во времени (например, развитие заболевания или результат лечения). Чтобы проверить эти прогнозируемые изменения, часто следует провести второе или третье исследование через пару дней или недель, опять же, по сути, ограничиваясь точечными измерениями.

Этот тип измерений всегда ограничен несколькими временными точками. По своей природе он не может дать какого-либо заключения о динамическом поведении физиологических систем. В этом случае функциональные нагрузочные пробы являются единственным, но при этом все еще ограниченным подходом к оценке динамического поведения. Благодаря такому подходу может быть достигнуто лучшее понимание функционального состояния пациента и сформулирован прогноз его изменений с учетом динамического поведения физиологической регуляции.

Биосигналы нуждаются в тщательной интерпретации хорошо подготовленными специалистами в различных областях медицины. Таким образом, комментарии и экспертные оценки записанных сигналов так же важны, как и сами необработанные цифровые данные. Только такой подход позволяет исследовать конкретные паттерны биосигналов и интерпретировать результаты их анализа.

В клинической практике могут применяться различные функциональные пробы. *Функциональные пробы* — это различные дозирован-

ные нагрузки или возмущающие воздействия (задержка дыхания, изменение положения тела на поворотном столе и др.), позволяющие объективно оценить функциональное состояние систем организма. К функциональным пробам предъявляют определенные критерии. В первую очередь проба должна быть нагрузочной, т. е. вызывать изменения в исследуемой системе организма. При этом желательно, чтобы нагрузки, вызываемые в ходе функциональной пробы, были эквивалентны нагрузкам в жизненных условиях.

Для того чтобы результаты исследований были надежными, функциональные пробы должны быть стандартизованы: результаты, полученные в ходе выполнения теста, должны быть воспроизводимыми при сохранении условий проведения теста и в случае неизменного функционального состояния испытуемого. Наконец, проба должна быть объективной и безвредной.

Общие требования, предъявляемые к проведению исследований с функциональными пробами

1. Обеспечение нормального микроклимата в помещении для тестирования: тихая комната, в которой поддерживается постоянная температура в пределах 18–25 °С.
2. В тестировании должен принимать участие минимум исследовательского персонала.
3. Перед началом исследования испытуемый в течение 5–10 мин должен адаптироваться к окружающим условиям.
4. Необходимо исключить возникновение звуковых, световых и других, не относящихся к исследованию, сигналов. До и во время исследования необходимо устранить помехи, приводящие к эмоциональному возбуждению испытуемого, исключить разговоры с ним, телефонные звонки, появление в помещении, где проводится исследование, посторонних лиц.
5. Аппаратура для исследований должна быть обязательно заземлена или, в случае автономного питания, быть безопасной и удовлетворять требованиям ГОСТ Р 50444 «Приборы, аппараты и оборудование медицинские. Общие технические условия» и ГОСТ Р 50267.0 «Изделия медицинские электрические. Общие требования безопасности. Электромагнитная совместимость».

6. Испытуемый должен иметь полную информацию об исследовании и дать в письменном виде добровольное согласие на участие в исследовании.
7. Необходимо проинструктировать испытуемого о проведении исследования с функциональными пробами.
8. При исследовании с функциональными пробами должен присутствовать врач или сотрудник, имеющий право оказания первой медицинской помощи, для стимуляции системы кровообращения и дыхания.
9. Необходимо вести протокол исследования с функциональными пробами.
10. При исследовании с функциональными пробами испытуемый должен, по возможности, не кашлять и не сглатывать слюну.

Классификация функциональных проб

1. По характеру воздействия

1.1. Функциональные пробы с дозированной физической нагрузкой.

Эти пробы позволяют получить объективные данные о функциональном состоянии сердечно-сосудистой системы и полезны в практическом отношении: они характеризуют восстановительные процессы, что дает информацию для оценки функциональной готовности испытуемого. Кроме того, по сдвигам частоты сердечных сокращений (ЧСС), артериального давления (АД) можно косвенно судить о характере реакции на нагрузку и даже выявить ранние нарушения работоспособности. Динамические исследования с использованием проб позволяют наблюдать за тренированностью, а также изучать характер адаптации ЧСС к меняющимся условиям среды, что позволяет исследователю дозировать нагрузку индивидуально каждому испытуемому.

Функциональные пробы с дозированной нагрузкой подразделяются на *одномоментные*, *двухмоментные* и *трехмоментные*.

К одномоментным пробам относятся:

- проба Мартинэ-Кушелевского

Методика проведения: в состоянии покоя анализируют ВСР в течение пяти мин и измеряют артериальное давление. Затем обследуемый выполняет 20 глубоких приседаний за 30 с с вытянутыми вперед

руками. После выполнения нагрузки анализируют ВСР в течение пяти мин, а в промежутке между 10 и 50 с измеряется артериальное давление.

- проба Котова — Дешина

Методика проведения: в состоянии покоя анализируют ВСР в течение пяти мин и измеряют артериальное давление. Затем испытуемый в течение трех мин выполняет нагрузку в виде бега на месте в темпе 180 шагов в минуту с высоким подниманием бедра. Для женщин и детей данная проба проводится 2 мин. После выполнения нагрузки испытуемый садится, и в течение пяти мин восстановительного периода анализируется ВСР, а в промежутке между 10 и 50 с измеряется артериальное давление.

- проба Руфье

Пробу Руфье используют для оценки адаптации сердечно-сосудистой системы к физической нагрузке, а также применяют как простой и косвенный метод для определения физической работоспособности.

Методика проведения: у испытуемого, находящегося в течение 5 мин в положении сидя, анализируют ВСР. Затем испытуемый выполняет нагрузку в виде 30 приседаний за 45 с. После нагрузки он садится, и в течение 5 мин анализируют изменения ВСР.

- Гарвардский степ-тест

Тест разработан в лаборатории по изучению утомления в Гарвардском университете под руководством *D. B. Dilla* в 1936 г. Вначале тест использовался для оценки физической работоспособности морских пехотинцев американской армии. Проба заключалась в подъемах на ступеньку высотой 20 дюймов (50,8 см) с частотой 30 раз в одну мин. Затем были разработаны его модификации с учетом пола и возраста обследуемых. В настоящее время гарвардский степ-тест используют для оценки физической работоспособности лиц, занимающихся физической культурой и спортом.

Двухмоментные функциональные пробы состоят из двух нагрузок и выполняются с небольшим интервалом отдыха. Например, тест PWC170 и 15-секундный бег в максимальном темпе дважды с интервалом отдыха между тестами в три мин.

Трехмоментная комбинированная проба С. П. Летунова позволяет разносторонне исследовать функциональную способность сердечно-сосудистой системы. Методика проведения: у испытуемого в состоянии покоя в положении сидя (после пяти мин отдыха) анализируют в течение пяти мин ВСР и измеряют артериальное давление на последнем этапе анализа ВСР. Затем испытуемый выполняет три нагрузки:

- 1) 20 приседаний за 30 с;
- 2) 15-секундный бег на месте, выполняемый в максимальном темпе;
- 3) трехминутный бег на месте в темпе 180 шагов в мин с высоким подниманием бедра.

1.2. Пробы с изменением условий внешней среды:

- проба с вдыханием воздуха с различным содержанием кислорода и углекислого газа;
- пробы в условиях измененной температуры внешней среды (в термокамере) или атмосферного давления (в барокамере);
- пробы при воздействии на организм линейного или углового ускорения (в центрифуге).

1.3. Пробы с изменением положения тела в пространстве с помощью поворотного стола:

- ортостатические;
- клиностатические.

Ортостатические пробы дают важную информацию об изменении кровотока при переходе тела из горизонтального в вертикальное положение: в этом случае в нижней его половине депонируется значительное количество крови, ухудшается венозный возврат крови к сердцу и, следовательно, уменьшается выброс крови (на 20–30 %). Компенсация этого неблагоприятного воздействия осуществляется, главным образом, за счет увеличения ЧСС. Важная роль принадлежит и изменениям сосудистого тонуса. Если он снижен, то уменьшение венозного возврата может быть столь значительным, что при переходе в вертикальное положение может развиться обморочное состояние в связи с резким ухудшением кровоснабжения мозга.

У спортсменов ортостатическая неустойчивость, связанная с понижением венозного тонуса, развивается крайне редко. Вместе с тем при проведении пассивной ортостатической пробы она может выявляться. Поэтому использование ортостатических проб для оценки функционального состояния организма спортсменов считается целесообразным.

При проведении ортостатической пробы изменяется активность симпатического отдела ВНС. Поэтому целесообразно в этом случае анализировать ВСР в положении лежа и по окончании первой минуты пребывания в вертикальном положении.

Клиностатическая проба применяется для оценки возбудимости парасимпатического отдела вегетативной нервной системы. Целесоо-

бразно при проведении этих исследований анализировать ВСР в течение пяти мин, когда испытуемый находится в положении стоя. Затем поворотный стол с испытуемым переводится в горизонтальное положение и в течение 300 с у него опять анализируется ВСР.

1.4. Пробы с использованием фармакологических и пищевых средств.

Используют с целью дифференциальной диагностики между нормой и патологией. По принципу фармакологического тестирования эти пробы принято делить на нагрузочные и пробы исключения. К нагрузочным относятся те пробы, в которых применяемый фармакологический препарат оказывает стимулирующее действие на исследуемый физиологический или патофизиологический механизм.

Пробы исключения основаны на ингибирующих (блокирующих) эффектах целого ряда препаратов.

1.5. Пробы с воздействием на внешнее дыхание:

- проба Вальсальвы;
- гипервентиляционная проба;
- гипоксические пробы (пробы Штанге, Генчи).

Проба Вальсальвы: в состоянии покоя анализируют ВСР в течение пяти мин в положении лежа. Затем испытуемый делает глубокий вдох и выдыхает воздух через мундштук (15–20 с) или сжатые губы (30–40 с). После выдоха испытуемый спокойно дышит. После выполнения нагрузки анализируют ВСР в течение пяти мин.

Гипервентиляционная проба: в состоянии покоя анализируют ВСР в течение пяти мин в положении лежа. Затем испытуемый должен сделать 20–30 форсированных глубоких вдохов и выдохов с большой частотой без перерыва в течение 20–30 с. После выполнения нагрузки анализируют ВСР в течение пяти мин.

Гипоксические пробы. Проба Штанге: измеряется максимальное время задержки дыхания после субмаксимального вдоха. Исследуемому предлагают сделать вдох, выдох, а затем вдох на уровне 85–95 % от максимального. При этом плотно закрывают рот и зажимают нос пальцами. Регистрируют время задержки дыхания.

Проба Генчи: регистрация времени задержки дыхания после максимального выдоха. Исследуемому предлагают сделать глубокий вдох, затем максимальный выдох. Исследуемый задерживает дыхание при зажатом пальцами носе и плотно закрытом рте. Регистрируется время задержки дыхания между вдохом и выдохом.

1.6. Тесты с психоэмоциональной нагрузкой.

Когнитивная нагрузка, стимулирующая активизацию работы лобных и теменных областей коры головного мозга: элементарные математические примеры.

Эмоциональная нагрузка: просмотр видео с изображением природы с аудио-сопровождением в виде спокойной инструментальной музыки.

2. По критерию оценки пробы

- Количественные — нагрузка и оценка пробы выражается какой-либо величиной.
- Качественные — оценка пробы ведется путем определения типа реакции сердечно-сосудистой системы на нагрузку (повысился, снизился или не изменился).

3. В зависимости от времени регистрации показателей

- Рабочие — показатели регистрируются в покое и непосредственно во время выполнения нагрузки.
- Послерабочие — показатели фиксируются в покое и после прекращения нагрузки в период восстановления.

Практическое задание к главе 2

1. Совместно с преподавателем проведите пробное исследование. Для этого включите один из приборов функциональной диагностики в соответствии с руководством его эксплуатации. Оцените качество и характер биосигналов в зависимости от условий съема, наличия (отсутствия) артефактов и функционального состояния испытуемого.

2. Проведите исследование с одной из функциональных проб в соответствии с общими требованиями, предъявляемыми к проведению исследований. Сохраните полученный результат. Результаты исследований будут использованы для дальнейшей обработки в главах 3 и 4.

Контрольные вопросы к главе 2

1. Дайте определение биомедицинскому сигналу.
2. Перечислите биомедицинские сигналы первой и второй группы.

3. Опишите артефакты, формируемые при регистрации биомедицинских сигналов.
4. Раскройте термин «электрокардиография»: опишите методику регистрации и исследования электрических полей, образующихся при работе сердца. Перечислите основные элементы электрокардиограммы.
5. Раскройте термин «электроэнцефалография»: опишите методику регистрации и исследования биоэлектрической активности головного мозга. Перечислите ритмы биоэлектрической активности головного мозга.
6. Опишите сигнал кожно-гальванической реакции: укажите физиологические особенности формирования и электрофизические характеристики.
7. Раскройте термин «стабилометрия»: опишите методику регистрации и исследования вестибулярной функции.
8. Сформулируйте диагностические особенности речевого сигнала.
9. Охарактеризуйте реографический сигнал: опишите методику регистрации и укажите информационные особенности при функциональных исследованиях.
10. Охарактеризуйте фотоплетизмографический сигнал: опишите методику регистрации и укажите информационные особенности при функциональных исследованиях.
11. Охарактеризуйте лазерную доплеровскую флоуметрию: опишите методику регистрации и укажите информационные особенности при функциональных исследованиях.
12. Охарактеризуйте пульсоксиметрию: опишите методику регистрации и укажите информационные особенности при функциональных исследованиях.
13. Охарактеризуйте ультразвуковую доплерографию сосудов головного мозга и шеи: опишите методику регистрации и укажите информационные особенности при функциональных исследованиях.
14. Перечислите общие требования и обеспечение безопасности при проведении функциональных исследований.
15. Перечислите функциональные нагрузочные пробы и охарактеризуйте их классификацию.

Глава 3.

Хранение биомедицинских сигналов^{*}

Биомедицинские базы данных широко используются в различных областях для множества целей: в инженерии — для разработки и тестирования программного обеспечения; в научных исследованиях — для валидации алгоритмов или методологий анализа данных; в здравоохранении и повседневной жизни людей [43].

Учитывая продолжающуюся технологическую революцию, получение биомедицинских сигналов перестало ограничиваться специализированной средой и дорогим оборудованием. Фактически получение этих данных теперь легко осуществить с помощью портативных и беспроводных устройств, что делает их широко распространенными [44].

Способ хранения данных может зависеть от цели, реквизитов или типа исследования, типа биомедицинских данных и устройства/производителя, использованного для сбора данных. Тот факт, что в настоящее время получен огромный объем данных, имеющих актуальную и значимую информацию для всех областей, о которых сообщается, усиливает необходимость структурировать и стандартизировать их, способствуя сотрудничеству и содействуя научным исследованиям [43; 44].

В настоящее время в нескольких общедоступных базах данных имеются все виды биомедицинских сигналов в свободном доступе. Некоторые из них имеют стандартные протоколы хранилищ данных и специальные API для доступа к данным, а другие — нет. Кроме того, некоторые базы данных категоризируют биомедицинские сигналы по типу, в то время как другие имеют несколько файлов данных, которые могут хранить несколько типов сигналов, находящихся в одном

^{*} Авторы главы 3: Gamboa Hugo, Rodrigues Joao (Universidade Nova de Lisboa, Lisbon), перевод А. Ю. Долганова (Уральский федеральный университет, г. Екатеринбург)

и том же файле, также сохраняя некоторые детали условий, в которых были получены сигналы (метаданные). В 2011 году было задокументировано 100 форматов [45].

3.1. Форматы биосигналов

В онтологии *BioSignal* [45] дается определение для описания биомедицинского сигнала: «любой вид измеряемой во времени величины, связанный с прямым результатом биологического процесса». Это последовательность изменяющихся во времени точек данных, которые были отобраны путем измерения биологического процесса с определенной частотой. Следовательно, сигнал имеет, по существу, выборки данных, которые хранятся в файле с определенным форматом. Каждый из них связан с моментом времени, который начинается с 0 и увеличивается для каждой выборки на основе определенного периода сбора данных. Кроме того, с образцами данных могут быть связаны *события* и *аннотации*.

Следуя номенклатуре определения *BioSignal*, *событие* — это событие во времени, указывающее, что с сигналом произошло что-то важное. *Аннотации* имеют время, связанное с ним, и касаются указания комментариев к конкретным событиям в сигнале.

Способ хранения образцов данных и их соответствующие форматы сильно различаются. *Запись*, которая определяется как *набор сигналов от одного сеанса записи*, может быть сохранена в одном файле или группе файлов вместе. Другие форматы могут включать события и/или аннотации, а также метаданные.

В этой главе описаны наиболее распространенные и известные форматы файлов, в частности, в области биомедицинских сигналов.

Требования и технические характеристики для хранения биосигналов

Рассматриваемые форматы файлов сигналов будут описаны в соответствии с набором требований и технических характеристик. Они основаны на том, что задокументировано в *Carlos Carreiras et. al.* [44]:

- 1) *производительность доступа*: оценивает скорость чтения и записи, а также тип доступа к данным и возможности сжатия данных в формате данных;

- 2) *межплатформенная поддержка*: учитывая широкий список форматов данных и то, что некоторые из них являются специфическими для типа данных, в то время как другие имеют конкретное происхождение базы данных, поддержка загрузки файлов или доступа к данным может быть ограничена, поэтому требуется межплатформенная поддержка при сравнении параметров;
- 3) *поддержка событий и аннотаций*: как упоминалось ранее, некоторые форматы данных выигрывают от присутствия событий и/или аннотаций в файле данных;
- 4) *расширяемость*: возможность добавлять больше данных в файл также является важной особенностью формата данных;
- 5) *метаданные*: объяснение или краткое изложение условий, в которых были получены данные, имеет первостепенное значение для более глубокого понимания типа используемых данных и может также использоваться в вычислительном отношении;
- 6) *мультимодальность*: способность файла хранить сигналы разной природы или разной частоты дискретизации;
- 7) *один файл*: данные, относящиеся к съему данных (сырые данные, события, аннотации, метаданные и т. д.), которые могут храниться в одном файле или в нескольких файлах.

Перечисленный набор функций используется для описания форматов файлов, и некоторые из них рассмотрены для сравнения между собой.

Формат текстового файла (TXT)

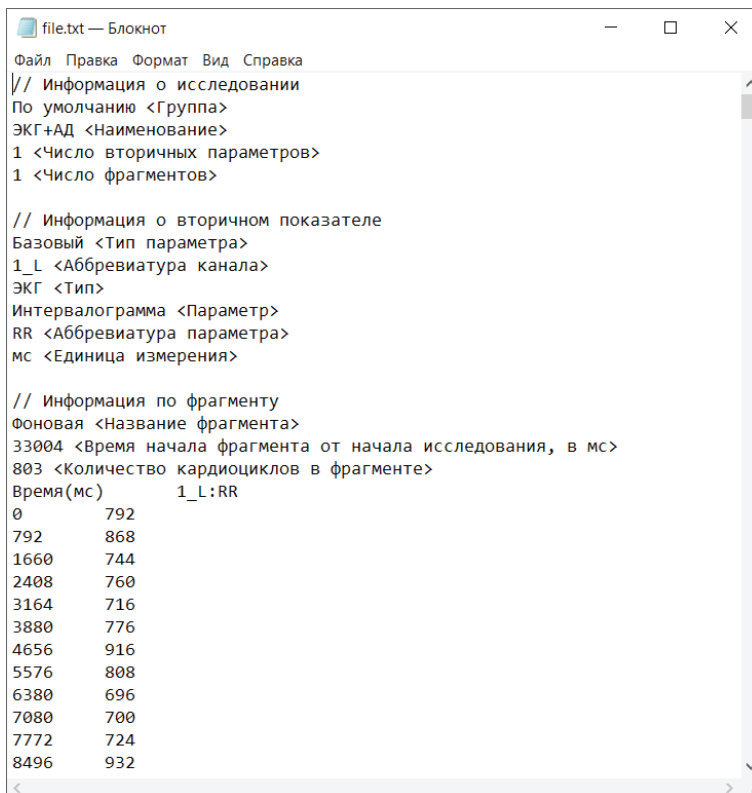
Простой файл *TXT* универсально используется для хранения выборок данных биомедицинского сигнала. Этот формат структурирован как последовательность строк электронного текста, который обычно кодируется из набора символов *ASCII* или *Unicode* (UTF-8). Преимуществами этого формата являются его простота и удобство доступа; тем не менее, он показывает некоторые ограничения в отношении емкости хранилища. Поскольку он имеет низкую энтропию, данные занимают больше места, чем необходимо. Таким образом, извлечение данных из файла происходит медленнее, чем в других форматах.

Файлы *TXT* могут быть структурированы в виде *значений, разделенных запятыми* (*.csv) или *значений, разделенных табуляцией* (*.tsv), если данные разделены запятой или табуляцией, соответственно. До-

бавление метаданных может быть сделано с добавлением текста в начале файла в качестве *заголовка*, но это зависит от решения производителя устройства.

Импортирование результатов

Ниже рассмотрим операции считывания файлов биомедицинских сигналов в формате *ТХТ* на примере текстового файла сигнала ВСР, зарегистрированного с использованием соответствующего канала записи электроэнцефалографа-анализатора «Энцефалан-131–03» (оборудования ООО научно-производственно-конструкторская фирма «Медиком МТД» г. Таганрог). Исходные данные ВСР представляют собой файл, содержащий сопутствующую информацию об исследовании и два столбца данных в мс: время регистрации текущего *R-R* интервала и длительность текущего *R-R* интервала (рис. 3.1).



```

file.txt — Блокнот
Файл  Правка  Формат  Вид  Справка
// Информация о исследовании
По умолчанию <Группа>
ЭКГ+АД <Наименование>
1 <Число вторичных параметров>
1 <Число фрагментов>

// Информация о вторичном показателе
Базовый <Тип параметра>
1_L <Аббревиатура канала>
ЭКГ <Тип>
Интервалограмма <Параметр>
RR <Аббревиатура параметра>
мс <Единица измерения>

// Информация по фрагменту
Фоновая <Название фрагмента>
33004 <Время начала фрагмента от начала исследования, в мс>
803 <Количество кардиоциклов в фрагменте>
Время(мс)      1_L:RR
0                792
792              868
1660             744
2408             760
3164             716
3880             776
4656             916
5576             808
6380             696
7080             700
7772             724
8496             932
  
```

Рис. 3.1. Исходный файл

Наиболее общим способом открытия является применение функции **open** (*file*, *mode*). Функция **open** возвращает объект файла, тип которого зависит от режима и посредством которого выполняются стандартные операции с файлами, такие как чтение и запись. Основным аргументом этой функции — *file* — является либо текстовой, либо байтовой строкой, задающей имя (и путь, если файл не находится в текущем рабочем каталоге) открываемого файла.

Аргумент *mode* является необязательной строкой, указывающей режим, в котором файл открывается. По умолчанию используется значение `'r'`, что означает «открыт для чтения в текстовом режиме». Другими общими значениями являются `'w'` — для записи (усечение файла, если он уже существует), `'x'` — для создания и записи в новый файл и `'a'` — для добавления в конец файла.

Откроем файл, используя менеджер контекста **with**, который гарантирует закрытие файла:

```
with open ('file.txt') as text:
    tempfile = text.readlines()
```

Метод **.readlines()**, применяющийся к объекту, который возвращает функция **open**, позволяет построчно считать поток информации и возвращает его в виде списка (рис. 3.2) строк.

Использование **.readlines()** считывает весь файл в память сразу. Если вы имеете дело с достаточно большим файлом, это может вызвать серьезные проблемы с производительностью или привести к сбою программы. Для открытия больших файлов рекомендуется итерационно присоединять каждую строку:

```
tempfile = []
with open ('ecg_data.txt') as text:
    for line in text:
        tempfile.append(line)
```

Как видно из примера (см. рис. 3.1–3.2) файлы биомедицинских сигналов содержат не только «сырые» данные биосигналов, но и метаданные (тип сигнала, условия съема и т. д.). Для непосредственной обработки метаданные не нужны. В таком случае необходимо извлекать только сырые данные. Для этого можно воспользоваться регулярными выражениями.

Модуль библиотеки **re** предоставляет операции сопоставления регулярных выражений, аналогичные тем, которые встречаются в Perl. Регулярное выражение определяет набор строк, которые ему соответ-

ствуют; функции в этом модуле позволяют проверить, соответствует ли конкретная строка заданному регулярному выражению (или соответствует ли заданное регулярное выражение определенной строке, что сводится к тому же самому). Для подключения библиотек в *Python* их необходимо «импортировать», используя команду **import**.

```

▼  tempfile = {list} <class 'list': [ '// Информация о исследовании\n', 'По умолчанию <Группа>\n',
01 000 = {str} '// Информация о исследовании\n'
01 001 = {str} 'По умолчанию <Группа>\n'
01 002 = {str} 'ЭКГ+АД <Наименование>\n'
01 003 = {str} '1 <Число вторичных параметров>\n'
01 004 = {str} '1 <Число фрагментов>\n'
01 005 = {str} '\n'
01 006 = {str} '// Информация о вторичном показателе\n'
01 007 = {str} 'Базовый <Тип параметра>\n'
01 008 = {str} '1_L <Аббревиатура канала>\n'
01 009 = {str} 'ЭКГ <Тип>\n'
01 010 = {str} 'Интервалограмма <Параметр>\n'
01 011 = {str} 'RR <Аббревиатура параметра>\n'
01 012 = {str} 'мс <Единица измерения>\n'
01 013 = {str} '\n'
01 014 = {str} '// Информация по фрагменту\n'
01 015 = {str} 'Фоновая <Название фрагмента>\n'
01 016 = {str} '33004 <Время начала фрагмента от начала исследования, в мс>\n'
01 017 = {str} '803 <Количество кардиоциклов в фрагменте>\n'
01 018 = {str} 'Время(мс)\t1_L:RR\n'
01 019 = {str} '0\t 792\n'
01 020 = {str} '792\t 868\n'
01 021 = {str} '1660\t 744\n'
01 022 = {str} '2408\t 760\n'
01 023 = {str} '3164\t 716\n'
01 024 = {str} '3880\t 776\n'
01 025 = {str} '4656\t 916\n'
01 026 = {str} '5576\t 808\n'

```

Рис. 3.2. Результат применения метода **readlines**

Помимо библиотеки *re* в примере ниже используется билиотека *numpy* Библиотека *numpy* — это фундаментальный пакет для научных вычислений на *Python*. Помимо очевидного научного использования, *numpy* также может использоваться как эффективный много-

мерный контейнер общих данных. Произвольные типы данных могут быть определены. Это позволяет `numpy` легко и быстро интегрироваться с широким спектром баз данных.

```
import numpy as np
import re
tempdata=np.array([re.findall(r'[-+]?[\d]*[\.]?[\d]+' ,
tf.replace(',',',.')) for tf in tempfile])
```

В указанном выше примере используется метод `re.findall` (*pattern*, *string*), находящий все элементы в строке *string*, которые соответствуют шаблону *pattern*. В нашем случае все строки проходят в цикле (`for tf in tempfile`), а шаблон можно расшифровать следующим образом:

- [группа_символов] соответствует любому одиночному символу, входящему в группа_символов;
- ? соответствует предыдущему элементу ноль или один раз;
- \d соответствует любой десятичной цифре;
- * соответствует предыдущему элементу ноль или более раз;
- + соответствует предыдущему элементу один или более раз;
- \. соответствует символу "." (разделяющий знак).

Описанное выше регулярное выражение будет соответствовать необязательному знаку, за которым следует либо ноль или более цифр, за которыми следуют точка и одна или несколько цифр (число с плавающей запятой с необязательной целочисленной частью), либо одна или несколько цифр (целое число), рис. 3.3.

Наконец, выделим только те строки, которые содержат два числа — это и будет выгруженным сигналом ВСП. Применение метода `.astype` ('float') позволяет создать массив, содержащий данные в формате числа с плавающей точкой (а не в формате строк), рис. 3.4.

```
td=np.array([len(d) for d in tempdata])
data = np.vstack(tempdata[td > 1]).astype('float')
```

Рассмотренный выше случай является достаточно общим и может применяться для большинства форматов данных. Однако в ряде случаев можно воспользоваться другими способами. Так, если файл данных не содержит метаданных и состоит только из чисел, то можно воспользоваться функцией чтения текстовых библиотеки `numpy.loadtxt`:

```
data = np.loadtxt("table.txt", delimiter='\t',
dtype=np.float)
```

```

▼ tempdata = {ndarray} [list()] list([]) list([]) list(['1'])
  01 min = {str} 'not a numeric object'
  01 max = {str} 'not a numeric object'
  > 1 2 shape = {tuple} <class 'tuple': (822,)
  > dtype = {dtype} object
  01 size = {int} 822
▼ array = {NdArrayItemsContainer} <pydevd_plug
  > 1 000 = {list} <class 'list': []
  > 1 001 = {list} <class 'list': []
  > 1 002 = {list} <class 'list': []
  > 1 003 = {list} <class 'list': ['1']
  > 1 004 = {list} <class 'list': ['1']
  > 1 005 = {list} <class 'list': []
  > 1 006 = {list} <class 'list': []
  > 1 007 = {list} <class 'list': []
  > 1 008 = {list} <class 'list': ['1']
  > 1 009 = {list} <class 'list': []
  > 1 010 = {list} <class 'list': []
  > 1 011 = {list} <class 'list': []
  > 1 012 = {list} <class 'list': []
  > 1 013 = {list} <class 'list': []
  > 1 014 = {list} <class 'list': []
  > 1 015 = {list} <class 'list': []
  > 1 016 = {list} <class 'list': ['33004']
  > 1 017 = {list} <class 'list': ['803']
  > 1 018 = {list} <class 'list': ['1']
  > 1 019 = {list} <class 'list': ['0', '792']
  > 1 020 = {list} <class 'list': ['792', '868']
  > 1 021 = {list} <class 'list': ['1660', '744']

```

Рис. 3.3. Результат применения регулярных выражений

```

▼ data = {ndarray} [[ 0. 792.]...View as Array
  01 min = {float64} 0.0
  01 max = {float64} 605664.0
> 1 2 3 shape = {tuple} <class 'tuple'>: (803, 2)
> dtype = {dtype} float64
  01 size = {int} 1606
▼ array = {NdArrayItemsContainer} <pydevd_plugins.extension
> 000 = {ndarray} [ 0. 792.]...View as Array
> 001 = {ndarray} [792. 868.]...View as Array
> 002 = {ndarray} [1660. 744.]...View as Array
> 003 = {ndarray} [2408. 760.]...View as Array
> 004 = {ndarray} [3164. 716.]...View as Array
> 005 = {ndarray} [3880. 776.]...View as Array
> 006 = {ndarray} [4656. 916.]...View as Array
> 007 = {ndarray} [5576. 808.]...View as Array
> 008 = {ndarray} [6380. 696.]...View as Array
> 009 = {ndarray} [7080. 700.]...View as Array
> 010 = {ndarray} [7772. 724.]...View as Array
> 011 = {ndarray} [8496. 932.]...View as Array
> 012 = {ndarray} [9432. 884.]...View as Array
> 013 = {ndarray} [10316. 832.]...View as Array
> 014 = {ndarray} [11148. 772.]...View as Array
> 015 = {ndarray} [11920. 752.]...View as Array
> 016 = {ndarray} [12676. 776.]...View as Array
> 017 = {ndarray} [13448. 820.]...View as Array
> 018 = {ndarray} [14264. 752.]...View as Array
> 019 = {ndarray} [15024. 884.]...View as Array
> 020 = {ndarray} [15904. 768.]...View as Array
> 021 = {ndarray} [16668. 804.]...View as Array

```

Рис. 3.4. Импортированный сигнал BCP

Сохранение результатов

В ходе обработки необходимо сохранять результаты. Используя функцию **open**, можно не только открывать файлы для чтения, но и для записи. Двумерные таблицы можно сохранять в текстовых файлах с помощью функции **.savetxt** библиотеки **numpy**:

```
with open('save.txt', 'wb') as file:
    np.savetxt(file, data, fmt=' %1.3f', delimiter='\t')
```

Европейский формат данных (EDF +)

Формат файла *EDF* является одним из наиболее широко используемых форматов для хранения биомедицинских сигналов, был основан в 1992 году и расширен как *EDF+* в 2003 году. Он поддерживает несколько языков программирования, таких как *Python*, *R*, *C*, *Matlab* и *Java* [46].

Внутренняя структура файла *EDF+* охватывает заголовок, за которым следуют записи данных. Заголовок содержит информацию о том, как был выполнен опыт, включая имя пациента, и поддерживает несколько записей данных, хранящихся в файле. Однако формат не позволяет хранить дополнительные, определенные пользователем, метаданные [45].

Что касается раздела записей данных, данные записываются в виде двухбайтовых целых чисел, и несколько биосигналов могут быть записаны одновременно. Все сигналы должны быть записаны с одинаковыми временными интервалами между выборками. Аннотации могут быть добавлены в файл с определенными временными выборками, которые не должны соответствовать частоте выборки [46].

Формат данных Биосеми (BDF)

Формат файла *BDF* — это расширенная версия формата файла *EDF*, имеющая дополнительный байт, что означает, что он может хранить данные до 24 бит, в то время как формат файла *EDF* может хранить данные до 16 бит [47]. Преимущества этого формата такие же, как формат *EDF*.

Для открытия для чтения/записи файлов *EDF/BDF* создана библиотека *PyEDFlib* с открытым исходным кодом.

```
import pyedflib
import numpy as np
f = pyedflib.EdfReader("data/test_generator.edf")
n = f.signals_in_file
signal_labels = f.getSignalLabels()
sigbufs = np.zeros((n, f.getNSamples()[0]))
```

```
for i in np.arange(n):  
    sigbufs [i,:] = f.readSignal (i)
```

Общий формат данных для биосигналов (GDF)

Формат файла *GDF* изначально был разработан для того, чтобы справиться с некоторыми ограничениями исходного формата файла *EDF* и разработать общий, гибкий и однозначный формат файла для всех существующих биосигналов, особенно для более сложных случаев, таких как область исследования интерфейс «мозг-компьютер» (*BCI*). [48].

Форматы файлов *GDF* поддерживают различные частоты дискретизации, общее кодирование событий, аннотации, кодирование настроек фильтра, несколько типов данных. Совсем недавно, с версией 2.2, были добавлены некоторые функции, которые по существу связаны с метаданными.

Общая структура содержит:

- 1) 256-битный фиксированный заголовок;
- 2) 256-битную переменную заголовка для каждого канала;
- 3) третий необязательный заголовок;
- 4) раздел данных, в котором хранится количество записей, умноженное на размер каждой записи в байтах;
- 5) таблицу событий, где каждое событие хранится с 6 или 12 байтами [49].

Иерархический формат данных (HDF)

HDF — это формат данных с самоописанием, который используется для хранения и управления большими и сложными объемами данных. Он поддерживается многими языками программирования, а именно *C*, *Java*, *Matlab* и *Python*. Первоначально был *HDF4*, но *HDF5* был разработан, чтобы справиться с некоторыми предыдущими ограничениями [50].

Файловая структура *HDF5* состоит из двух основных типов объекта:

- 1) наборы данных, которые являются многослойными иерархическими заголовками объектов с информацией о типе данных, пространстве данных, разметке, среди других свойств данных, указывающих на необработанные данные сигнала;

2) группы, являющиеся заголовком объекта, который может содержать наборы данных и другие группы.

Способ хранения информации аналогичен файлам *JSON*, что делает этот формат гибким и позволяет добавлять точки аннотаций и события, а также метаданные, связанные с наборами данных или объектами группы [50].

На языке *Python* работа с форматом *HDF5* реализована при помощи библиотеки *h5py*. Это позволяет хранить огромные объемы числовых данных и легко использовать над этими данными операции из библиотеки *numpy*. Например, можно нарезать наборы данных размером в несколько терабайт, хранящиеся на диске, как если бы они были настоящими массивами *numpy*. Тысячи наборов данных могут быть сохранены в одном файле, классифицированы и помечены.

```
import h5py
filename = 'file.hdf5'
f = h5py.File(filename, 'r')
# Выводим на экран список групп, содержащийся в файле
print("Keys: %s" % f.keys())
a_group_key = list(f.keys())[0]
# считывание данных
data = list(f[a_group_key])
#!/usr/bin/env python

# Создание массива случайных данных
import numpy as np
data_matrix = np.random.uniform(-1, 1, size=(10, 3))
# Запись файла в формат HDF5
data_file = h5py.File('file.hdf5', 'w')
data_file.create_dataset('group_name', data=data_matrix)
data_file.close()
```

Формат MAT-файла (MAT)

Формат файла *MAT* был специально создан *Matworks* для языка программирования *Matlab*. Он сохраняет массивы данных и матрицы в структурированном виде.

MAT-файл подразделяется на заголовок (128 байт), за которым следует один или более элементов данных. Каждый элемент состоит из тега и элемента данных. Тег имеет два 32-битных поля, определяющих тип данных и количество битов, что указывает на то, как их следует интерпретировать.

Заголовок содержит первые 116 байтов для текстовых данных, удобных для восприятия человеком, предоставляя информацию о том, как был создан файл, но не о самой записи. Остальные байты заголовка являются структурной информацией, которая может понадобиться при загрузке файла *MAT*. Поскольку *MAT*-файл поддерживает массивы и матрицы, он может хранить несколько типов сигналов и изображений.

Большинство типов данных поддерживаются в *MAT-file*, и *API* были разработаны для использования *MAT*-файлов на языках программирования, отличных от *Matlab*, таких как *Python*, *R*, *Fortran* и *C* [51]. Так, для *Python* импортное использование *MAT*-файлов реализовано в библиотеке *scipy.io*

```
import scipy.io
mat = scipy.io.loadmat('file.mat')
```

Формат файла базы данных *Physionet* (WFDB)

В базе данных *Physionet* каждая запись хранится в трех разных файлах:

- 1) заголовок (*.hea*);
- 2) файл двоичного сигнала (*.dat*);
- 3) файл аннотации (*.atr*) [52].

Файл *.dat* содержит оцифрованные образцы одного или нескольких сигналов. Файл *.hea* описывает метаданные, связанные с тем, как были получены сигналы, и некоторой дополнительной информацией о сигналах. Аннотации также включены в файл *.atr*, содержащий наборы меток, каждая из которых описывает особенность одного или нескольких сигналов в указанное время в записи.

Формат *.dat* специально структурирован базой данных *Physionet*, но его можно открыть с помощью специального набора инструментов под названием *WFDB* из *Physionet*. Пример взаимодействия с базой данных *Physionet* будет подробно описан в п. 3.3, с. 74.

Расширяемый формат данных (XDF)

Формат файла *XDF* является контейнерным форматом общего назначения для данных многоканальных временных рядов с обширной связанной метаинформацией. Он имеет открытый исходный код и предназначен для данных биосигналов, таких как ЭЭГ, ЭМГ, ЭОГ, ЭКГ, КГР, но он также может обрабатывать данные с высокой частотой дискретизации (например, аудио) или данные с большим количеством каналов (например, *fMRI* или видео). Метаданные хранятся в формате *XML* [53].

Формат *XDF* относительно похож на *EDF*, но заголовок преобразуется в *XML*, а данные хранятся в двоичном разделе. Есть *API* для *Python* и *Matlab* [53].

Биорас — формат файла AcqKnowledge (ACQ)

Формат файла *ACQ* для *Windows/PC* был создан специально компанией *Biopac Systems Inc.*, Голета, Калифорния, США. Записи хранятся изначально в формате *ACQ*. Этот формат доступен для чтения только в системе *Biopac*, но данные можно экспортировать в текстовый файл или использовать *API* для *C++* для более глубокого анализа данных [54].

Спецификация формата *ACQ* общедоступна и используется в системах *Biopac*. Формат включает в себя:

- 1) секцию заголовка, которая содержит информацию, касающуюся захвата, такую, как количество полученных каналов и частота дискретизации;
- 2) секцию данных канала, где конкретный канал характеризуется числом, смещением и масштабом амплитуды, количеством выборок и единиц измерения;
- 3) раздел типа данных;
- 4) секцию данных канала, где данные хранятся в чередующемся формате.

NI — Техническое решение для управления данными (TDMS)

Формат файла *TDMS* — это формат файла *National Instruments* для сохранения данных измерений в хорошо документированном виде.

Структура формата файла является иерархической и состоит из файла, группы и канала.

Каждый файл может содержать неограниченное количество групп, группы могут содержать неограниченное количество каналов. Файлы содержат метаданные и необработанные данные. Метаданные описательны и хранятся в виде *XML*, а массивы данных, связанные с объектами канала, называются необработанными данными. Файлы *TDMS* содержат необработанные данные для нескольких каналов в одном непрерывном блоке. Эти данные извлекаются с использованием индекса необработанных данных, который включает в себя информацию о составе блока данных, включая канал, который соответствует данным, количество значений, которые блок содержит для этого канала, и порядок, в котором данные были сохранены [55].

Другие форматы

Можно упомянуть другие форматы файлов, а именно формат ЭМГ, более детально разработанный компанией *Delsys* и используемый для хранения ЭМГ и сигналов движения; формат *DCR* от системы сбора данных *AstroNova*, разработанный этой компанией и имеющий определенный формат данных для записи биосигналов; *HL7aECG*, который был разработан, чтобы иметь формат файла для аннотированного хранения ЭКГ, который бы соответствовал требованиям Американского управления по контролю за продуктами и лекарствами (*FDA*) и требованиям *HL7*; формат файла ЭЭГ от компании *Compumedics Neuroscan*, разработанный для записи сигналов и изображений ЭЭГ; формат файла *C3D*, специально разработанный для трехмерных биомеханических сигналов [56]; Формат файла *MFER*, разработанный в Японии для записи нескольких биосигналов [57]; и *MP3*, *WMV* и *WAV* формат файла для акустических биосигналов.

Сравнение форматов файлов

Некоторые из перечисленных форматов файлов были сравнены с точки зрения свойств, определенных в начале главы 3. Таблица следует описанию, приведенному в [44], используя следующие обозначения: (?) Информация не найдена о свойстве; (—) свойство отсутству-

ет; (+) свойство присутствует; (++) свойство выражено хорошо; (+++) свойство выражено очень хорошо.

Таблица

Сравнение форматов файлов данных для биосигналов

Свойство	EDF +	BDF +	GDF +	WFDB	HDF5	MAT	TXT	TDMS	ACQ
скорость доступа	—	—	+	—	+++	++	—	++	++
межплатформенная поддержка	++	++	++	++	++	+	+++	+	+
поддержка событий и аннотаций	+	+	+	++	++	—	—	++	++
расширяемость	—	—	—	—	+++	++	+	?	?
метаданные	+	++	++	+	+++	+	+	+++	++
мультимодальность	+	++	+	+	+++	++	—	+	+
один файл	+	+	+	—	+	+	+	+	+

Данные в таблице отражают различия между свойствами форматов файлов, хотя несколько форматов файлов имеют очень похожие характеристики. Формат файла с лучшими свойствами — *HDF5*. Он показывает несколько преимуществ в отношении других файлов, таких как скорость доступа к ним, а также наличие метаданных.

3.2. Базы данных биосигналов

Базы данных биосигналов разрабатываются с конкретными целями, будь то для конкретных проектов, типов анализа или исследования или просто для ссылки на конкретный биосигнал. Эти базы данных должны объяснять протоколы сбора данных, количество субъектов и их соответствующие характеристики для анализа сигналов. В Интернете можно найти несколько баз данных, а именно наиболее известную базу данных *Physionet*. Такие базы данных необходимы для научных разработок, исследований в области обработки биомедицинских сигналов и в области здравоохранения.

В этом пункте описаны найденные базы данных, объяснены основные назначения этих баз данных, включенные проекты и тип хранимых биосигналов со ссылками на ссылки для доступа к набору дан-

ных. Для тех баз данных, которые поддерживает *API*, будет дана краткая справка о том, как извлечь биосигналы.

Многие базы данных находятся в онлайн-репозиториях с конкретными биосигналами, которые были приобретены для определенного исследования, проекта, конкурса или другой цели. Есть онлайн-базы данных с репозиториями сигналов из нескольких источников, таких как хорошо известный репозиторий *Physiobank*, архив классификации временных рядов Университета Калифорнии (*UCR*), репозиторий машинного обучения из Университета Калифорнии в Ирвине (*UCI*) и репозиторий *Kaggle*. Репозиторий *PhysioBank* специфичен для физиологических сигналов, в то время как остальные репозитории включают в себя все виды временных рядов, но имеют конкретные случаи, когда биосигналы доступны для задач машинного обучения.

Physionet

Physionet описывается как «большой и растущий архив хорошо охарактеризованных цифровых записей физиологических сигналов и связанных с ними данных для использования сообществом биомедицинских исследований» [52]. Архив содержит более 75 наборов данных, которые можно бесплатно загрузить, охватывающих несколько типов биосигналов, таких как многопараметрические сердечно-легочные, нервные и другие биомедицинские сигналы от здоровых испытуемых и пациентов с различными состояниями, имеющими серьезные последствия для общественного здравоохранения, включая внезапную сердечную смерть, застойную сердечную недостаточность, эпилепсию, нарушения походки, апноэ во сне и старение.

Архив временных рядов для классификации UCR

Архив временных рядов *UCR*, представленный в 2002 году, стал важным ресурсом сообщества интеллектуального анализа временных рядов, помог в публикации тысячи статей, используя по крайней мере один набор данных из архива. Первоначальное воплощение архива имело шестнадцать наборов данных, но с тех пор он подвергался периодическим расширениям. В настоящее время в архиве имеется 128 наборов данных, которые свободно и легко доступны [58].

Доступные наборы данных специфичны для проблем интеллектуального анализа данных, каждый из которых разделен на две категории: *TRAIN* и *TEST*. Хотя в архиве имеется широкий спектр типов сигналов, можно найти несколько специализированных наборов данных биосигналов, а именно: ЭКГ, ЭКГ плода, жесты и движения человека, электромагнитная артикулография, ЭЭГ и речевой сигнал.

Хранилище Машинного Обучения UCI

Репозиторий машинного обучения *UCI* описывается как «набор баз данных, теорий предметной области и генераторов данных, которые используются сообществом машинного обучения для эмпирического анализа алгоритмов машинного обучения». Архив был создан как в 1987 году Дэвидом Ага и его аспирантами в Калифорнийском университете в Ирвине. Он широко используется образовательными и исследовательскими сообществами в качестве источника наборов данных для машинного обучения. Он имеет более 475 наборов данных всех типов временных рядов и изображений. Можно найти несколько наборов данных биосигналов, а именно для ЭКГ, ЭЭГ, акустических и речевых сигналов, сигналов движения жестов и повседневной жизнедеятельности [59].

Kaggle

Kaggle — это веб-сообщество ученых и энтузиастов данных. Платформа используется для обмена открытыми наборами данных, участия в соревнованиях по машинному обучению или обмена кодом в среде данных. При таком подходе *Kaggle* имеет большой выбор наборов данных, в том числе несколько с биосигналами. Большинство наборов данных связано с распознаванием человеческой деятельности, то есть с данными, записанными с помощью инерционных датчиков. Другие наборы данных также найдены, а именно для данных ЭЭГ, фотоплетизмографии и артериального давления. Большинство наборов данных находится в формате *CSV* и может быть загружено напрямую.

Другие наборы данных

Как упомянуто выше, есть несколько наборов данных, которые можно найти в онлайн-хранилищах для конкретных исследовательских

проектов, онлайн-соревнований или других целей. Некоторые наборы данных принадлежат больницам, консорциумам или образовательным учреждениям, поэтому для загрузки доступных данных необходимо заполнить форму запроса. Данные из представленных баз данных поступают в различных форматах, таких как *TXT*, *MAT*, *HDF5* и другие.

Наиболее распространенные доступные наборы данных связаны с исследованиями электрокардиографии и электроэнцефалографии. Можно представить два соответствующих проекта, таких как набор данных *DEAP*, в котором 32 участника записали свои сигналы ЭЭГ при просмотре 40 различных одночасовых видеофильмов с музыкой в качестве фона, для изучения аффективных состояний человека [60]; и Консорциум исследований безопасности сердца, который после принятия доставляет базу данных хранилища ЭКГ из *FDA* с тысячами записей ЭКГ [61].

3.3. API для баз данных и анализа биосигналов

Несколько баз данных поддерживают API для доступа и обработки данных. В этом пункте описан набор инструментов, доступных в онлайн-хранилищах.

Physionet

Для *Physionet* существует несколько вспомогательных инструментов с целью:

- 1) визуализации данных;
- 2) интеллектуального анализа данных;
- 3) преобразования данных *Physionet* в стандартные форматы (*TXT*, *CSV*, *EDF*, *MAT* и *ZIP*);
- 4) создания записей, совместимых с *Physionet*;
- 5) анонимизации медицинских карт в файлах *Physionet* и *EDF*;
- 6) обработки сигналов;
- 7) частотного анализа временных рядов;
- 8) нелинейного анализа временных рядов;
- 9) анализа специфических сигналов ЭКГ;
- 10) физиологические модели и симуляции.

Как правило, пакетное программное обеспечение *WFDB* имеет большинство из этих утилит, но другие решения также могут применяться [52].

Программный пакет *WFDB* — это библиотека *WFDB*, приложения для обработки сигналов и программное обеспечение *WAVE* для визуализации данных, аннотирования и интерактивного анализа данных формы волны. Программные *API* работают на *C*, *C++* и *Fortran*, но существуют программные пакеты для *Python* и *Matlab*, которые можно использовать для записи и чтения сигналов и аннотаций *wfdb*. Хранилище *Physionet* также обеспечивает поддержку оболочек *Python*, *Java*, *C#* и *Matlab* для полного спектра возможностей библиотеки *WFDB*, написанной на *C++*.

Для того чтобы получить полный список наборов данных, к которым есть доступ, необходимо воспользоваться функцией `get_dbs()`. В основном наборы данных в *Physionet* делятся на два типа:

- 1) клинические наборы данных — данные из клинических учреждений интенсивной терапии, которые могут включать демографические данные, измерения показателей жизнедеятельности, проводимые у постели больного, результаты лабораторных анализов, процедуры, лекарства, заметки лиц, осуществляющих уход, изображения и отчеты о снимках, а также данные о смертности (как в больнице, так и вне ее);
- 2) наборы данных *Waveform* — непрерывные записи физиологических сигналов с высоким разрешением. Базы данных сигналов организованы в соответствии с их типами сигналов и аннотаций.

Если вам известен конкретный набор данных, с ним можно взаимодействовать напрямую. В первую очередь можно загрузить файлы из базы данных на персональный компьютер, используя функцию `dl_database(db_dir, dl_dir, records)`, где название набора данных, из которого будут скачены файлы, — директория, в которую эти файлы будут загружены, список названий файлов, которые необходимо загрузить. Полный список названий файлов, которые содержатся в наборе данных, можно получить, используя функцию `get_record_list(db_dir)`. Наконец считывание загруженного файла в *Python* происходит с использованием функции `rdrecord(record_name)`, где `record_name` — имя файла, который необходимо считать (содержит полный путь к файлу). Ниже приведен пример загрузки и считывания первого файла из набора дан-

ных `mgfdb`. Применение метода позволяет обращаться к конкретным записанным сигналам (которые различаются в зависимости от набора данных). Так, в приведенном примере первый сигнал соответствует каналу ЭКГ, а четвертый канал — инвазивному датчику артериального давления.

```
import wfdb
files=wfdb.get_record_list('mgfdb')
wfdb.dl_database(db_dir='mgfdb',
dl_dir='C:/physionet/',
records= [files[0],])
record = wfdb.rdrecord('C:/physionet/'+ files[0])
ECG = record.p_signal[:, 0]
AP = record.p_signal[:, 3]
```

Классификация временных рядов UCR

Нет *API* для доступа к данным или анализа данных. Тем не менее, веб-сайт предоставляет все алгоритмы, разработанные в контексте классификации временных рядов в *метакоде* и *Java*, к которым можно получить доступ в репозитории *bitbucket* [62]. Эта информация в основном актуальна для разработчиков.

Kaggle

В хранилище *Kaggle* есть не только данные, но и код, и дискуссионные форумы, касающиеся анализа и классификации данных. Они могут быть опробованы онлайн с *ядрами Kaggle*, написанными сценариями на *R* или *Python*.

Другие открытые решения

В *Python* есть несколько наборов инструментов для анализа данных, которые можно использовать в качестве руководства для загрузки биосигналов, анализа физиологических данных и классификации данных. Примерами являются *BioSSPy* и *BiosignalsNotebooks*.

Набор инструментов *BioSSPy* [63] имеет различные методы обработки сигналов и распознавания образов для анализа биосигналов. Он содержит модули для биометрии, кластеризации, построения графиков,

метрик расстояния и хранения. Он также содержит набор примеров сигналов, таких как *BVP*, *ECG*, *EDA*, *EMG* и *Respiration*.

BiosignalsNotebooks [64] — это набор *Jupyter Notebooks*, написанных на *Python*, которые служат учебными пособиями для загрузки, обработки, обнаружения событий, анализа данных, извлечения параметров и поддержки машинного обучения. На странице представлена библиотека сигналов для тестирования. Этот набор учебных пособий также поддерживает процедуры сбора данных, поскольку они связаны с программным инструментом *OpenSignals* от *Plux*.

Существует также библиотека программного обеспечения с открытым исходным кодом для обработки биомедицинских сигналов под названием *BioSig* [65]. Он имеет набор инструментов обработки сигналов для различных областей применения, таких как нейроинформатика, компьютерно-мозговые интерфейсы, нейрофизиология, психология, сердечно-сосудистые системы и исследования сна. Кроме того, анализ биосигналов для сигналов ЭЭГ, ЭКГ, ЭОГ, ЭМГ или дыхания является очень важным элементом проекта *BioSig*. Существуют решения для сбора данных, обработки артефактов, контроля качества, извлечения признаков, классификации, моделирования и визуализации данных.

Инструментом для визуализации биосигналов является *SigViewer* [66]. Это приложение с открытым исходным кодом для просмотра биосигналов и создания, редактирования и отображения информации о событиях (например, аннотации или выбор артефактов).

Практическое задание к главе 3

1. Используя результаты исследований, сохраненные при выполнении практического задания к главе 2, проведите импорт сигналов в среду *Python*. Обработка импортированных сигналов будет проводиться при выполнении практических заданий к главе 4.
2. Проанализируйте содержание наборов данных в базе *Physionet*. Совместно с преподавателем выберете набор данных, который представляет наибольший интерес. Проведите скачивание и импортирование набора данных. Обработка отдельных сигналов будет проводиться при выполнении практических заданий к главе 4. Совместная обработка сигналов будет проводиться при выполнении практических заданий к главе 5.

Контрольные вопросы к главе 3

1. Перечислите требования и технические характеристики для хранения биосигналов.
2. Укажите форматы биосигналов, которые являются модификацией формата *EDF*.
3. Укажите формат биомедицинских сигналов, который состоит из нескольких файлов.
4. Перечислите основные преимущества формата *HDF5*.
5. Перечислите базы данных биосигналов с открытым доступом.
6. Опишите алгоритм получения сигналов из базы данных *Physionet*.

Глава 4.

Расчет параметров биомедицинских сигналов^{*}

К начальным этапам анализа биосигналов относят первичный визуальный анализ биосигнала, предварительную обработку биосигнала и расчет значимых параметров биомедицинских сигналов. Визуальный анализ позволяет качественно оценить чистоту зарегистрированного сигнала, оценить возможные артефакты записи и дать общую информацию о сигнале. Предварительная обработка связана с устранением артефактов биосигналов и интерполяцией неэквидистантных биосигналов. Для расчета значимых параметров биосигналов традиционно используют широкий спектр методов: от статистического и анализа гистограмм распределения до спектрального анализа и нелинейных методов.

4.1. До обработки сигналов

Визуализация

Одной из важных составляющих обработки является визуализация полученных результатов. Для этого необходимо воспользоваться функциями библиотеки `matplotlib.pyplot`. `Matplotlib` — это библиотека *Python 2D* для построения графиков, гистограмм, спектров мощности, диаграмм ошибок, диаграмм рассеяния и т.д. с помощью всего лишь нескольких строк кода. Для простой прорисовки модуль `pyplot` предоставляет интерфейс, похожий на *MATLAB*. Для опытного пользователя имеется полный контроль над стилями линий,

^{*} Автор главы 4: А. Ю. Долганов (Уральский федеральный университет, г. Екатеринбург)

свойствами шрифта, свойствами осей и т.д. с использованием объектно-ориентированного интерфейса или набора функций, знакомых пользователям *MATLAB*.

Создаем элемент типа **figure** (рисунок) и задаем его размеры (в дюймах).

```
import matplotlib.pyplot as plt
fig=plt.figure()
fig.set_figheight(20)
fig.set_figwidth(40)
```

Функция `plt.plot(x, y, [fmt])` используется для построения зависимости y от x в виде линий и/или маркеров. Координаты точек или линейных узлов задаются как x , y . Необязательный параметр *fmt* — это способ определения базового форматирования, такого как цвет, маркер и стиль линии. Параметр *fmt* может задаться в полной форме, прописывая каждый параметр отдельно (цвет, тип маркера, тип линии), либо в сокращенной. Так, выполнение следующих строк дает идентичные результаты.

```
plt.plot (data[:,0]/1000, data[:,1],
color='black', marker='o', linestyle='dashed',
linewidth=2, markersize=12)
plt.plot(data[:,0]/1000, data[:,1], 'ko -',
linewidth=2, markersize=12)
```

Создаем подписи к осям и название графика, используя функции `plt.xlabel`, `plt.ylabel` и `plt.title` соответственно. Ниже `label_font` и `ticks_font` — объекты типа «Словарь», в которых содержатся параметры шрифта (название и размер). Функциями `plt.xticks` и `plt.yticks` задается формат значений осей x и y . Функция `plt.grid` добавляет сетку на рисунок. Применение функции `plt.tight_layout()` компоует содержимое элемента **figure** таким образом, чтобы минимизировать пустые места и пересечения содержимого.

```
label_font = {'fontname':'Times New Roman',
'size':'35'}
ticks_font = {'fontname':'Times New Roman',
'size':'25'}
plt.xticks(**ticks_font)
plt.yticks(**ticks_font)
plt.xlabel('Время, с', **label_font)
```

```
plt.ylabel('R-R, mc', **label_font)
plt.grid()
plt.tight_layout()
```

Отображение полученного рисунка реализуется с помощью команды `plt.show()`. Результат выполнения описанных выше команд показан на рис. 4.1.

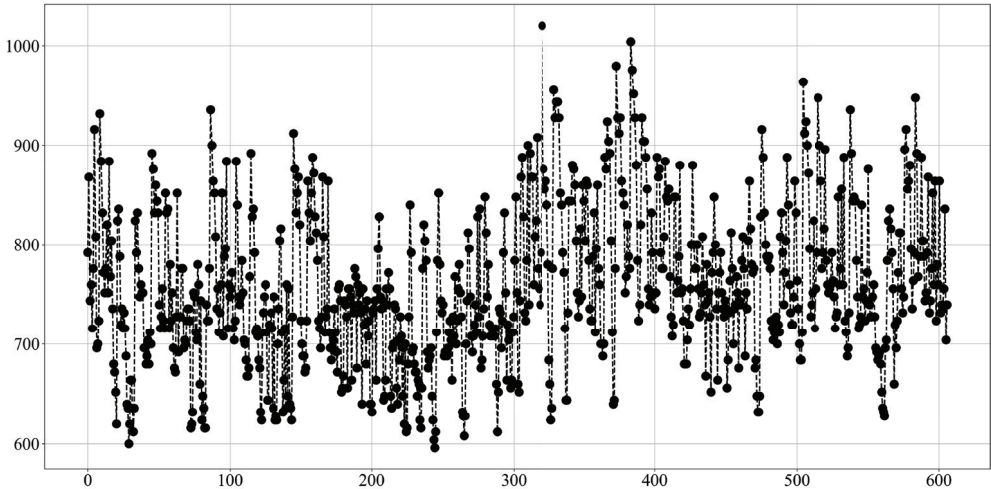


Рис. 4.1 Результат выполнения команды `plt.show()`

Для отображения двумерных объектов (таблицы) можно воспользоваться функцией `plt.imshow(X, cmap=None, norm=None, aspect=None, interpolation=None, alpha=None, vmin=None, vmax=None)`.

```
plt.imshow(numbers, cmap='jet', aspect='auto')
plt.colorbar()
```

Пример использования функции `plt.imshow` представлен в п. 4.6.

Сохранение результатов

В ходе обработки необходимо сохранять результаты. Используя функцию `open`, можно не только открывать файлы для чтения, но и для записи. Двумерные таблицы можно сохранять в текстовых файлах с помощью функции `np.savetxt`, библиотеки `numpy`.

```
with open('save.txt', 'wb') as file:
    np.savetxt(file, data, fmt='%1.3f', delimiter='\t')
```

В случае, если результаты состоят из более сложных элементов (кортежи), то можно сохранять результат в формате `npz`, нативном для библиотеки `numpy`, с помощью функции **savez**.

```
np.savez('outfile', data=data)
```

Для того чтобы импортировать данные обратно, необходимо воспользоваться функцией **load**.

```
file=np.load('outfile.npz')  
data=file['data']
```

Для сохранения изображений можно воспользоваться функцией `plt.savefig(fname, dpi, format)`:

- параметр *fname* — путь, или файл-подобный объект *Python*. Если формат не установлен, то выходной формат выводится из расширения *fname*, если оно есть, в противном случае форматом по умолчанию является **.png*;
- параметр *dpi* (возможные значения параметра `None`, число > 0 , `'figure'`) — разрешение в точках на дюйм. Если `None`, то берется формат по умолчанию. Если `'figure'`, использует значение *dpi* рисунка;
- параметр *format* — формат файла, например, `'png'`, `'pdf'`, `'svg'`.

Используя модуль *PdfPages*, имеем возможность сохранять несколько рисунков в один файл *pdf*.

```
from matplotlib.backends.backend_pdf import PdfPages  
with PdfPages('filename.pdf') as pdf:  
    pdf.savefig(fig)  
    plt.close(fig)
```

4.2. Предобработка сигналов

Фильтрация

Биомедицинские сигналы, как правило, проявляют себя как слабые сигналы. При этом шумом (артефактом, помехой) может считаться любой другой сигнал. В ходе записи биомедицинских сигналов могут возникнуть артефакты, связанные с механическими движениями тела во время съема, помехами в сети и наводками электромагнитного поля.

Всего существуют как минимум пять способов к устранению артефактов [67]:

- 1) синхронное усреднение/усреднение по ансамблю (в случае если имеется множество реализаций/копий сигналов);
- 2) фильтрация по методу скользящего среднего;
- 3) фильтрация в частотной области;
- 4) оптимальная фильтрация по Винеру;
- 5) адаптивная фильтрация.

Для сигналов ЭКГ, КГР, ЭЭГ наиболее применима фильтрация в частотной области, поскольку, как правило, шумы являются случайными процессами, которые статистически не зависят от сигнала, а спектр биомедицинских сигналов ограничен по полосе частот по сравнению со спектром шумов.

В языке *Python* функции фильтрации реализованы в модуле `signal` библиотеки `scipy`. Первым делом необходимо задать параметры фильтра, используя функцию `signal.firwin(numtaps, cutoff, pass_zero, fs)`, где:

- `numtaps`: длина фильтра (количество коэффициентов, т. е. порядок фильтра+1). `numtaps` должно быть нечетным, если полоса пропускания включает частоту Найквиста;
- `fs`: частота дискретизации сигнала (необязательный параметр);
- `cutoff`: частота среза фильтра (выраженная в тех же единицах, что и `fs`) или массив частот среза (то есть краев полосы). В последнем случае частоты в «отсечке» должны быть положительными и монотонно увеличиваться между 0 и $fs/2$. Значения 0 и $fs/2$ не должны быть включены в `cutoff`;
- `pass_zero` (`True` или `False`): если `True`, усиление на частоте 0 (т. е. «усиление постоянного тока») равно 1. В противном случае усиление постоянного тока равно 0.

Для сигналов ЭКГ рекомендуется применять частотную фильтрацию в диапазоне частот от 2 до 70 Гц. Нижняя граница обусловлена необходимостью удаления постоянного тренда (изолинии). Верхняя граница связана с электрическими наводками. В настоящий момент нет строгих предписаний, какая должна быть выбрана длина фильтра, однако для сигналов ЭКГ рекомендуется использовать фильтры высоких порядков (8, 10 и т. д.). Создадим фильтр для сигнала ЭКГ с частотой дискретизации 250 Гц (частота, используемая в соответствующем канале записи электроэнцефалографа-анализатора «Энцефалан-131–03»). Результат применения фильтра показан на рис. 4.2.

```
from scipy import signal
filter=signal.firwin(10,[2,70], fs=250, pass_zero=False)
```

Функция `signal.lfilter(b, a, x)` фильтрует сигнал с помощью фильтров с конечной и бесконечной импульсной характеристикой. Здесь b — вектор коэффициентов числителя (которые были получены с применением функции `signal.firwin()`), a — коэффициенты знаменателя, x — сигнал, который фильтруется.

```
filtered=signal.lfilter(filter, 1.0, ecg)
```

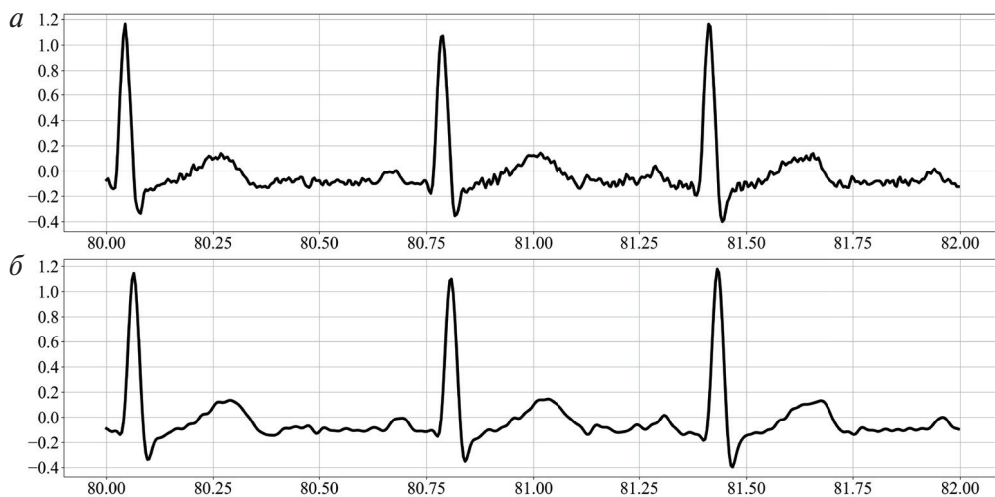


Рис. 4.2. Сигнал ЭКГ:

a — до фильтрации; b — после фильтрации

Эвристическая «починка» по методу «трех сигм»

Для сигналов ВСП артефакты могут появиться в ходе применения алгоритмов нахождения зубцов R . Как правило, под артефактами подразумеваются отклонения значений длительности R - R интервалов, которые не входят в диапазон ± 3 СКО от среднего значения M . Алгоритм очищения сигнала ВСП от артефактов представлен ниже. Здесь и далее `trr` — исходный ряд R - R интервалов, `NN` («normal to normal») означает ряд R - R интервалов без артефактов. Также в алгоритме удаляются те R - R интервалы, которые по длительности короче 300 мс и длиннее 1400 мс (т. к. эти значения маловероятны):

```

trr = trr[trr > 300]
trr = trr[trr < 1400]
M, O = np.mean(trr), np.std(trr)
index=0
NN=[]
timeNN=[]
for rr in trr:
    if (rr<M+3*O) and (rr>M-3*O) :
        NN.append(rr)
        if index==0:
            timeNN.append(0)
        else:
            timeNN.append(timeNN[index-1]+NN[index-1])
        index = index + 1
timeNN=np.array(timeNN)
NN=np.array(NN)

```

Интерполяция

Как правило, биомедицинские сигналы являются эквидистантными, поскольку регистрируются с определенной частотой дискретизации. Однако есть такой сигнал ВСП, который не является эквидистантным. Поэтому перед оценкой спектральных и нелинейных характеристик сигнала ВСП необходимо провести его интерполяцию, т.е. спроектировать на равномерную временную сетку. В языке *Python* интерполяция реализована в модуле `interpolate` библиотеки `scipy`.

Ниже представлена функция, которая возвращает равномерную временную сетку и интерполированный сигнал. Аргументы этой функции: `time` — временная сетка исходного сигнала, `signal` — значения исходного сигнала на временной сетке, `D` — желаемый шаг равномерной временной сетки.

```

from scipy import interpolate
def get_interpolation(time, signal, D):
    start= time[0]
    stop = time[-1]
    num=(stop-start)//D
    time_int = np.linspace(start, stop, num)
    fint = interpolate.interpld(time, signal, 'cubic')

```

```

    signal_int = fint(time_int)
    return time_int, signal_int

```

В начале функции задается равномерная временная сетка `time_int` с использованием команды `linspace(start, stop, num)`. Здесь `start` — начальное значение последовательности, `stop` — конечное значение последовательности, `num` — количество образцов для генерации.

Далее создается объект `fint`. Согласно [10] при анализе сигналов ВСР рекомендуется использовать сплайновую кубическую интерполяцию.

Наконец, к объекту `fint` применяется операция вызова с аргументом `time_int`. Операция вызова возвращает `signal_int` — интерполированный сигнал.

Практическое задание к п. 4.1 и 4.2

Выполните визуальный анализ и предобработку биосигнала, импортированного при выполнении практического задания к главе 3:

- 1) проведите импорт биосигнала;
- 2) визуализируйте оригинальный биосигнал;
- 3) выполните необходимую предварительную обработку биосигнала;
- 4) визуализируйте предобработанный биосигнал;
- 5) сохраните изображения и данные для дальнейшей обработки.

4.3. Статистические параметры

Статистические параметры применяются для прямой количественной оценки временных рядов биомедицинских сигналов. Для расчета этих параметров можно воспользоваться функциями библиотеки `numpy` и модуля `stats` библиотеки `scipy`.

Среднее значение M равняется математическому ожиданию случайной величины и показывает относительное расположение распределения на числовой прямой (первый статистический момент).

$$M = \frac{1}{N} \sum_{i=1}^N N N_i,$$

где N — количество элементов временного ряда NN , NN_i — i -й отсчет этого ряда;

```
M = np.mean (NN)
```

Среднеквадратическое отклонение (СКО, SD) сигнала (второй статистический момент) на наблюдаемом временном отрезке и показывает разброс распределения вокруг среднего значения. Численно равен корню дисперсии распределения:

$$SD = \sqrt{\frac{1}{N} \sum_{i=1}^N (NN_i - M)^2}$$

```
SD = np.std (NN)
```

Коэффициент асимметрии *skewness* является числовой характеристикой симметрии распределения (пропорционален третьему статистическому моменту). Коэффициент асимметрии положителен, если правый хвост распределения длиннее левого, и отрицателен в противном случае. Если распределение симметрично относительно математического ожидания, то его коэффициент асимметрии равен нулю.

```
from scipy import stats
skewness = stats.skew(NN, bias=False)
```

Коэффициент эксцесса *kurtosis* распределения показывает, насколько ярко выражена вершина распределения в окрестности среднего. Коэффициент эксцесса стандартного нормального распределения равен нулю. Он положителен, если пик распределения около математического ожидания острый, и отрицателен, если пик очень гладкий.

```
kurtosis = stats.kurtosis (NN, bias=False)
```

CV — коэффициент вариации, отношение SD к среднему значению, выраженное в процентах

$$CV = \frac{SD}{M} \cdot 100 \%$$

Частота пересечения нуля (ZCR) — это частота изменения знака сигнала, то есть частота, с которой сигнал изменяется с положительного через ноль на отрицательный или с отрицательного через ноль на положительный. В общем случае этот параметр может быть оценен для пересечения любого значения. Так, для сигнала ВСП интерес представляет частота пересечения среднего значения.

Для расчета этого параметра необходимо:

- 1) вычислить вектор разниц между исходным временным рядом и константы Z , частоту пересечения которой необходимо оценить;

- 2) поэлементно вычислить знак полученного временного ряда, используя команду `np.sign`;
 - 3) вычислить дискретную разницу полученного временного ряда;
 - 4) вычислить число нулевых значений дискретных разниц;
 - 5) выполнить нормировку на длину временного ряда:
- $$ZCR = (100 * \text{len}(\text{np.sum}(\text{np.diff}(\text{np.sign}(\text{np.asarray}(\text{NN}) - Z)) == 0)) / (\text{len}(\text{NN})))$$

Особые параметры для ВСР

Для сигнала ВСР также выделяют ряд специфичных статистических параметров, которые учитывают взаимосвязь между последующими интервалами [68].

RMSSD, мс — находится среднее значение суммы квадратов разности, последовательность отсчетов временного ряда *NN*. Из полученного выражения выделяется квадратный корень. Данный параметр связывают с краткосрочными изменениями в ВСР, и он соответствует компонентам ВСР, контролируемым блуждающим нервом.

$$RMSSD = \left[\frac{1}{N} \sum_{i=1}^{N-1} (NN_{i+1} - NN_i)^2 \right]^{0.5},$$

`RMSSD = np.sqrt(sum(pow(np.diff(NN), 2)) / len(NN))`

NN50 — количество пар последовательных отсчетов временного ряда *NN* на интервале наблюдения, различающихся более чем на 50 мс. Этот параметр также связан с краткосрочными изменениями и оценивает высокочастотные колебания в сигнале ВСР.

`NN50 = len(np.diff(NN)[abs(np.diff(NN)) > 50])`

pNN50, % — отношение оценки *NN50* к общему количеству пар последовательных отсчетов временного ряда *NN*, является инвариантной по отношению к длине временного ряда.

`pNN50 = 100.0 * len(np.diff(NN)[abs(np.diff(NN)) > 50]) / len(np.diff(NN))`

SDSD — стандартное отклонение разностей между соседними интервалами

`SDSD = np.sqrt((sum(np.diff(NN)) - sum(np.diff(NN))) / (len(NN) - 1)) ** 2 / (len(NN) - 2))`

Практическое задание к п. 4.3

Выполните статистический анализ биосигнала, импортированного при выполнении практического задания к главе 3:

- 1) произведите расчет статистических параметров биосигнала;
- 2) сохраните полученный результат.

4.4. Гистограммы распределения

Гистограмма является точным представлением распределения числовых данных. Чтобы построить гистограмму, первым шагом является бин (от англ. *bin*) для диапазона значений временного ряда, то есть разделение всего диапазона значений на ряд интервалов, а затем подсчет количества значений, попадающих в каждый интервал. Бины обычно указываются как последовательные непересекающиеся интервалы значений данных. Бины (интервалы) должны быть смежными и часто (но не обязательно) одинакового размера.

Если ячейки имеют одинаковый размер, над бином устанавливается прямоугольник с высотой, пропорциональной частоте — количеству значений, попадающих в заданный бин. Гистограмма также может быть нормализована для отображения «относительных» частот. В этом случае гистограмма показывает долю случаев, которые попадают в каждую из нескольких категорий с суммой высот, равной 1.

Гистограммы дают грубое представление о плотности базового распределения данных и часто используются для оценки функции плотности вероятности изучаемой переменной.

Особенности построения гистограмм для сигналов ВСП

Вариационные (или геометрические) параметры основаны на анализе закона распределения R - R интервалов как случайных величин. Для этого строятся гистограммы распределения R - R интервалов в координатах: «количество R - R интервалов K » — «длительность R - R интервалов τ » [68; 69]. Рекомендуемая ширина бина гистограммы — 50 мс [70]. Из всего многообразия гистограмм распределения ВСП наиболее типичными являются [71]:

- нормальная гистограмма, близкая по виду к кривым Гаусса, характерная для здоровых людей в состоянии покоя;
- асимметричная — указывает на нарушение стационарности процесса, наблюдается при переходных состояниях
- эксцессивная — характеризуется очень узким основанием и заостренной вершиной, регистрируется при выраженном стрессе, патологических состояниях;
- полимодальная — обусловлена наличием несинусового ритма (мерцательная аритмия, экстрасистолия), а также множественными артефактами при регистрации ВСР.

Для построения гистограммы в *Python* необходимо воспользоваться соответствующей командой библиотеки *matplotlib*.

```
fig = plt.figure()
trr=data[:,1]
n=max([np.round((max(trr)-min(trr))/50),1])
plt.hist(trr, n)
plt.show()
```

Основными тремя показателями являются мода, амплитуда моды и вариационный размах.

Мода M_0 , мс — для исследуемого временного ряда (ВР) это такое значение (R - R) интервала, которое наиболее часто встречается на интервале наблюдения. Данный параметр физиологически трактуется как наиболее вероятное значение кардиоритма на интервале наблюдения. При нормальном распределении временного ряда R - R интервалов M_0 мало отличается от математического ожидания M .

Амплитуда моды AM_0 , % — это число R - R интервалов, соответствующих значению моды, в % к объему выборки ВР. Физиологическая трактовка этого показателя связана со степенью активации симпатического отдела ВНС, которая обуславливает стабилизирующий эффект централизации управления ритмом сердца.

Вариационный размах VR , мс — отражает степень вариативности значений R - R интервалов в исследуемом временном ряде. Он вычисляется по разности максимального и минимального значений R - R интервалов. При наличии аритмий, асистола и/или артефактов этот показатель может быть искажен. В соответствии с рекомендациями при вычислении VR следует отбрасывать маловероятные значения R - R интервалов, если соответствующая высота бина гистограммы составля-

ет менее 3 % от общего количества R - R интервалов в рассматриваемом временном ряде. В физиологическом смысле показатель VR связывают с активностью парасимпатического отдела ВНС.

Для расчета трех основных показателей гистограммы необходимо воспользоваться следующими командами:

```
am, m=np.histogram(trr, int(n))
am=100*am/sum(am)
AM0=max(am)
Nmax=np.argmax(am)
M0=m[Nmax]
Y=m[am>3]
VR=max([Y[-1]-Y[0], 50.0])
```

Три вышеупомянутых показателя используются для составления ряда вторичных индексов, определенных согласно рекомендациям.

Рассмотрим вторичные индексы подробнее.

- ИН, у.е. — индекс напряжения регуляторных систем, который отражает «степень централизации управления ритмом сердца» [27]. В нормальных условиях определяется соотношением влияний симпатического и парасимпатического отдела вегетативной нервной системы. В ряде исследований параметр используется для объективизации эффективности терапии тревожных расстройств у пациентов с повышенной тревожностью

$$\text{ИН} = \frac{AM_0}{2M_0 * VR}.$$

- ИВР, у.е. — индекс вегетативного равновесия — может быть использован для определения соотношения между активностями симпатического и парасимпатического отделов ВНС. В норме значения этого показателя находятся в интервале (100–300) у.е, а повышенные значения свидетельствуют об увеличении тонуса симпатического отдела

$$\text{ИВР} = \frac{AM_0}{VR}.$$

- ВПР, у.е. — вегетативный показатель ритма, показатель, который оценивает смещение вегетативного баланса. Небольшие значения ВПР — 6 у.е. и меньше — указывают на преобладание парасимпатической активности

$$\text{ВПР} = \frac{1}{M_0 * VR}.$$

- ПАПР, у. е. — показатель адекватности процессов регуляции, который оценивает «степень централизации управления ритмом» [27] и соответствий изменений уровня функционирования синусового узла (функции автоматизма) на изменение симпатической активности

$$\text{ПАПР} = \frac{AM_0}{M_0}$$

Практическое задание к п. 4.4

Выполните анализ гистограммы распределения биосигнала, импортированного при выполнении практического задания к главе 3:

- 1) визуализируйте гистограмму распределения биосигнала;
- 2) сохраните гистограмму распределения биосигнала;
- 3) произведите расчет параметров гистограммы биосигнала;
- 4) сохраните полученный результат.

4.5. Спектральные показатели (Фурье-анализ)

Спектральный анализ применяется для точной количественной оценки периодических процессов в сердечном ритме. Физиологический смысл спектрального анализа состоит в том, что с его помощью оценивается активность отдельных уровней управления ритмом сердца. Спектральный метод позволяет качественно и количественно оценить соотношения частотных компонент сигнала.

Разработаны алгоритмы быстрого и эффективного расчета дискретного Фурье-преобразования — т. н. быстрое преобразование Фурье. Ниже представлена функция, которая возвращает вектор значений спектральной мощности и соответствующий вектор частот. Аргументы этой функции — интерполированный сигнал *signal* и частота дискретизации *D*.

```
def get_spectral_fourier(signal, D):
    l=(nextpow2((signal.size)))
    furie=np.fft.fft((signal-np.mean(signal)),
    n=l)/len(signal)
    Fs=1/D
    freq=(Fs/2)*np.linspace(0,1,(1+l/2))
    furie=furie[0: len(freq)]
    return freq, furie

freq, furie = get_spectral_fourier(signal, D)
```

При анализе биомедицинских сигналов интерес представляет оценка спектральной мощности в определенных полосах частот. Выбор конкретного диапазона частот зависит от биомедицинского сигнала.

Спектральный анализ ВСР

Спектральный анализ применяется для получения количественной оценки периодических процессов в сердечном ритме. При спектральном анализе сигналов ВСР выделяют следующие спектральные компоненты согласно российским и западным стандартам [27; 72]:

- диапазон *High Frequency HF* — от 0,40 до 0,15 Гц;
- диапазон *Low Frequency LF* — от 0,15 до 0,04 Гц;
- диапазон *Very Low Frequency VLF* — от 0,040 до 0,003 Гц.

При анализе данных холтеровских исследований (суточный мониторинг) дополнительно вводят диапазон ультранизкочастотных исследований (*ULF — Ultra Low Frequency*), содержащий частоты ниже 0,003 Гц.

Мощность высокочастотной составляющей спектра *HF* отражает активности парасимпатического отдела ВНС и автономного контура регуляции. Высокочастотные колебания сердечного ритма в спектре ВСР связаны с актом дыхания [73–76].

Мощность низкочастотной составляющей спектра *LF*, по мнению Р. М. Баевского, характеризует преимущественно состояние симпатической регуляции сосудистого тонуса [77–79].

Мощность очень низкочастотной составляющей спектра *VLF* обусловлена влиянием на ритм сердца надсегментарных вегетативных структур («центрального» уровня регуляции), поскольку амплитуда

этих волн тесно связана с психоэмоциональным напряжением и функциональным состоянием коры головного мозга [77; 80; 81].

Биологические ритмы, лежащие в том же диапазоне частот, что и *VLf* колебания ВСР, связаны с механизмами терморегуляции, колебаниями периферического сосудистого тонуса, активности ренина, секреции лептина [82], что позволяет предполагать участие этих механизмов в генезе *VLf*-колебаний. Существуют данные об увеличении мощности *VLf* при поражениях ЦНС, тревожно-депрессивных расстройствах [83].

В качестве спектральных показателей используются следующие величины:

- абсолютная спектральная мощность (mc^2) в каждом частотном диапазоне *HF* (*Fr*), *LF* (*Fr*) и *VLf* (*Fr*);

```
f=(0.4,0.15,0.04,0.003)
```

```
f1, f2, f3, f4=f[0], f[1], f[2], f[3]
```

```
HF=(np.power(abs(furie[(freq>f2)*(freq<f1)]),2))
```

```
HF_fr=sum(HF)
```

```
LF=(np.power(abs(furie[(freq>f3)*(freq<f2)]),2))
```

```
LF_fr=sum(LF)
```

```
VLf=(np.power(abs(furie[(freq>f4)*(freq<f3)]),2))
```

```
VLf_fr=sum(VLf)
```

- суммарная мощность спектра *Total Power* — *TP* (*Fr*) (mc^2);

```
TP_fr=HF_fr+LF_fr+VLf_fr
```

- нормированное значение мощности (%) в каждом частотном диапазоне в процентах от суммарной мощности спектра *TP* — *HF_n* (*Fr*), *LF_n* (*Fr*) и *VLf_n* (*Fr*);

- максимум значения спектра в каждом частотном диапазоне *HF_{max}*, *LF_{max}*, *VLf_{max}* и соответствующие значения частот *f* (*HF_{max}*), *f* (*LF_{max}*), *f* (*VLf_{max}*)

```
HFmax=max(HF)
```

```
freq_hf=freq[(freq>f2)*(freq<f1)]
```

```
f_HFmax=freq_hf[np.argmax(HF)]
```

```
LFmax=max(LF)
```

```
freq_lf=freq[(freq>f3)*(freq<f2)]
```

```
f_LFmax=freq_lf[np.argmax(LF)]
```

```
VLfmax=max(VLf)
```

```
freq_vlf=freq[(freq>f4)*(freq<f3)]
```

```
f_VLfmax=freq_vlf[np.argmax(VLf)]
```

Также выделяют ряд вторичных индексов, использующих комбинации показателей мощностей HF , LF и VLF :

- индекс вегетативного баланса — $\frac{LF}{HF}$;
- индекс централизации ИЦ

$$\text{ИЦ} = \frac{HF + LF}{VLF};$$

- индекс активации подкорковых нервных центров ИАП

$$\text{ИАП} = \frac{LF}{VLF}.$$

Спектральный анализ сигналов ЭЭГ

Спектральный анализ распространен и при обработке сигналов ЭЭГ. Принципиально изложенный выше подход может быть применен и к сигналам ЭЭГ, при условии соответствующих корректировок аргументов функций. Так, при использовании функции **get_spectral_fourier** необходимо подставлять соответствующее значение шага дискретизации D (зависит от аппарата для съема) и соответствующие диапазоны частот (дельта, альфа, тэта, бета и гамма ритмы).

Практическое задание к п. 4.5

Выполните Фурье-анализ биосигнала, импортированного при выполнении практического задания к главе 3:

- 1) выполните Фурье-преобразование биосигнала;
- 2) визуализируйте Фурье-спектр биосигнала;
- 3) сохраните Фурье-спектр биосигнала;
- 4) произведите расчет параметров Фурье-спектра биосигнала;
- 5) сохраните полученный результат.

4.6. Спектральные показатели (вейвлет-преобразование)

Для анализа нестационарных сигналов применяется вейвлет-преобразование, которое позволяет выявлять частотно-временные неоднородности исследуемых процессов как стационарных, так и нестационарных. Вейвлет-преобразование (wt) может быть использовано в качестве альтернативы Фурье (Fr) преобразованию [84]. Вместо гармонических функций, используемых при Фурье-преобразовании, при вейвлет-преобразовании используются математические функции определенной формы, которые локальны во времени и частоте.

Этот подход основан на расчете вейвлет-коэффициентов по формуле непрерывного вейвлет-преобразования:

$$W(a, b) = \frac{1}{\sqrt{a}} \int s(t) \cdot \psi\left(\frac{t-b}{a}\right) dt,$$

где a — масштабирующий параметр, b — параметр сдвига, ψ — анализирующий базисный вейвлет, $s(t)$ — исходный сигнал.

Связь масштабирующего параметра a с частотой определяется следующим соотношением:

$$a = \frac{f_c}{D \cdot f}.$$

где f_c — центральная частота вейвлета, D — шаг дискретизации, f — исследуемая частота.

Под мощностью в частотном диапазоне $[f_1: f_2]$ подразумевается квадрат соответствующей частотной составляющей. Частотная составляющая $s_{f_1, f_2}(t)$ — временной ряд, полученный обратным вейвлет-преобразованием согласно следующему соотношению:

$$s_{f_1, f_2}(t) = C_{\psi}^{-1} \int_{a_1}^{a_2} \int W(a, b) \cdot \psi\left(\frac{t-b}{a}\right) \frac{da db}{a^2},$$

где a_1 и a_2 — масштабирующие параметры, соответствующие частотам f_1 и f_2 , C_{ψ}^{-1} константа, характерная для базисной функции.

Ниже представлена функция, которая возвращает матрицу прямого вейвлет-преобразования (`cwt`), матрицу обратного вейвлет-преобразования (`icwt`) и соответствующий вектор частот. Аргументы этой

функции — интерполированный сигнал, название базисной вейвлет-функции (w), нижняя и верхняя граница частотного диапазона ($f1$, $f2$) и частота дискретизации D . Эта функция использует библиотеку PyWavelets. Библиотека сочетает в себе простой интерфейс высокого уровня с низким уровнем C и производительностью *Cython*.

```
import pywt
def get_wavelets(signal, w, f1, f2, D):
    fc = pywt.central_frequency(w)
    a1=fc/(D*f1)
    a2=fc/(D*f2)
    lna1=np.log(a1)
    lna2=np.log(a2)
    ln_a=np.linspace(lna1, lna2,100)
    A=np.exp(ln_a)
    F=fc/(D*A)
    dA=np.zeros(A.shape)
    dA=np.diff(A)
    dA=dA.tolist()
    dA.append(dA[-1])
    cwt, frq = pywt.cwt(signal - np.mean(signal), A, w)
    icwt = np.transpose(np.multiply(np.transpose(cwt),
    dA/pow(A,1.5)))
    return cwt, icwt, F
```

```
cwt, icwt, F = get_wavelets(signal, w, f1, f2, D)
plt.imshow(np.abs(cwt), cmap='jet', aspect='auto')
```

Результат визуализации матрицы вейвлет-преобразования показан на рис. 4.3.

Для оценки спектральных составляющих необходимо просуммировать матрицу обратного вейвлет-преобразования по соответствующим индексам. Временные сигналы, получаемые в результате обратного вейвлет-преобразования, показаны на рис. 4.4.

```
f=(0.4,0.15,0.04,0.003)
fnum1=int(0)
fnum2=np.argmax(F>f[1])[-1][0]
fnum3=np.argmax(F>f[2])[-1][0]
fnum4=int(len(F))
```

```
HFt= sum(V[range(fnum1, fnum2)])
LFt= sum(V[range(fnum2, fnum3)])
VLFt= sum(V[range(fnum3, fnum4)])
```

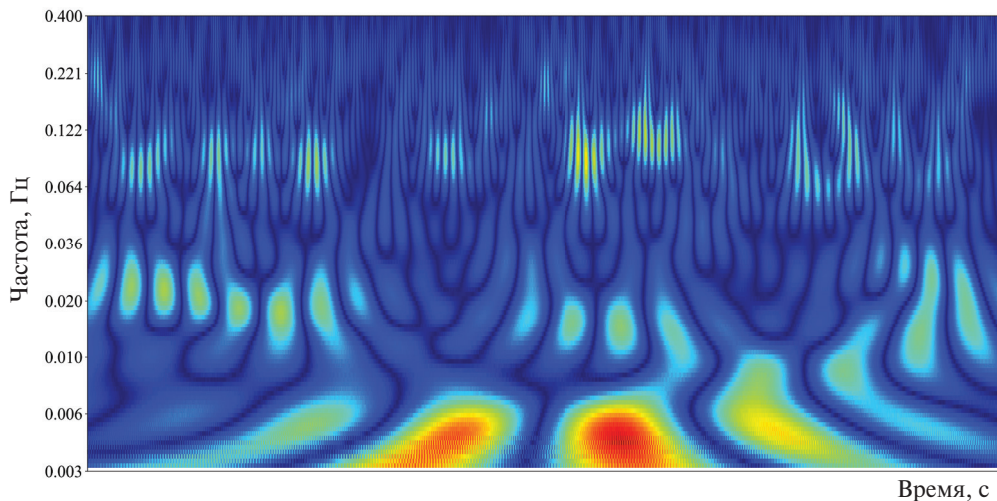


Рис. 4.3. Матрица вейвлет-преобразования

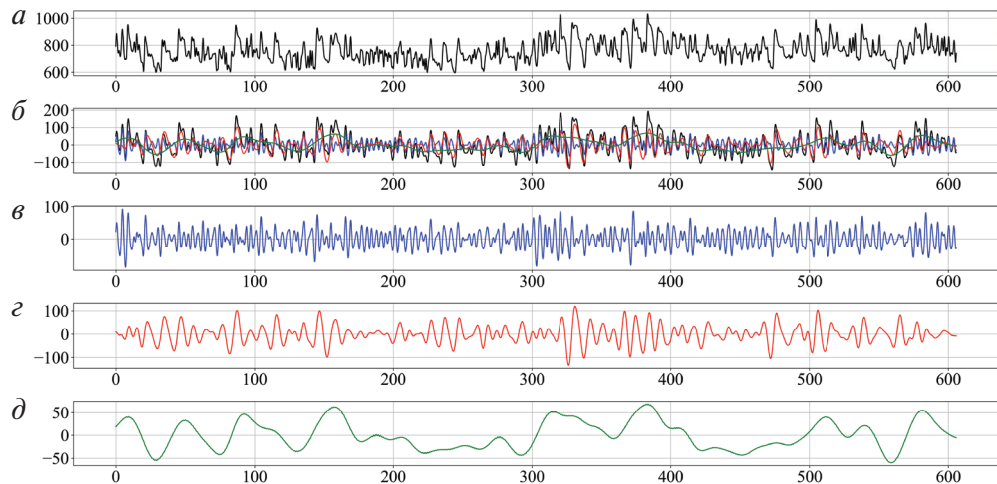


Рис. 4.4. Обратное вейвлет-преобразование сигнала ВСП:

a — исходный сигнал; *б* — суммарный сигнал отдельных частотных компонент;
в — высокочастотная компонента *HFt*; *г* — низкочастотная компонента *LFt*;
д — очень низкочастотная компонента *VLFt*

С помощью вейвлет-преобразования можно рассчитывать аналогичные базовые показатели:

- абсолютную спектральную мощность (mc^2) в каждом частотном диапазоне $HF(wt)$, $LF(wt)$ и $VLF(wt)$;
- суммарную мощность спектра *Total Power* — $TP(wt)$ (mc^2);
- нормированное значение мощности (%) в каждом частотном диапазоне в процентах от суммарной мощности спектра TP — $HF_n(wt)$, $LF_n(wt)$ и $VLF_n(wt)$;
- индекс вегетативного баланса — $LF/HF(wt)$.

Помимо этого в качестве характеристик ВСР можно оценивать стандартные отклонения $SDHF$, $SDLF$, $SDVLF$ временных рядов $HF(t)$, $LF(t)$, $VLF(t)$ интенсивностей периодических составляющих ВСР в диапазонах частот HF , LF и VLF соответственно, а также стандартное отклонение $SDTP$, интенсивности суммарной мощности спектра.

Дополнительно особенности вейвлет-преобразования ВСР можно анализировать по данным временного ряда индекса вегетативного баланса $F[(LF(t))/(HF(t))]$ [85]. Таким образом, в качестве параметров локальных дисфункций AF функционала $F[(LF(t))/(HF(t))]$ применяют количество дисфункций Nd , максимальное значение их амплитуды $(LF/HF)_{max}$ и интенсивность в интервале наблюдения $(LF/HF)_{int}$.

Практическое задание к п. 4.6

Выполните анализ биосигнала с использованием вейвлет-преобразования, импортированного при выполнении практического задания к главе 3:

- 1) выполните вейвлет-преобразование биосигнала;
- 2) визуализируйте матрицу вейвлет-преобразования биосигнала;
- 3) сохраните матрицу вейвлет-преобразования биосигнала;
- 4) выполните обратное вейвлет-преобразование биосигнала;
- 5) визуализируйте компоненты обратного вейвлет-преобразования биосигнала;
- 6) сохраните изображение компонент обратного вейвлет-преобразования биосигнала;
- 7) произведите расчет параметров вейвлет-преобразования биосигнала;
- 8) сравните результаты при использовании различных базисных функций;
- 9) сохраните полученный результат.

4.7. Нелинейные оценки

Энтропия Шеннона

Информационная энтропия — это средняя скорость, с которой информация генерируется стохастическим источником данных. Мера информационной энтропии, связанной с каждым возможным значением данных, является отрицательным логарифмом функции вероятности массы для значения:

$$En = -\sum_i p_i \log p_i,$$

где p_i — вероятность i -го значения данных.

Когда источник данных выдает значение с низкой вероятностью (то есть когда происходит событие с малой вероятностью), событие несет больше «информации» («неожиданности»), чем когда данные источника выдают значение с высокой вероятностью. Количество информации, передаваемой каждым событием, определенным таким образом, становится случайной величиной, ожидаемое значение которой является информационной энтропией. Обычно энтропия относится к беспорядку или неопределенности, и определение энтропии, используемое в теории информации, прямо аналогично определению, используемому в статистической термодинамике.

Ниже представлена функция для *Python*, которая сначала рассчитывает частоту встречаемости каждого элемента в массиве `data`, а затем, используя соответствующую функцию библиотеки `scipy.stats`, вычисляет энтропию массива `data`.

```
import numpy as np
from scipy.stats import entropy

def count_entropy(data, base=None):
    value, counts = np.unique(data, return_counts=True)
    return entropy(counts, base=base)
```

Approximate Entropy

В статистике аппроксимированная энтропия (*ApEn*) — это метод, используемый для количественной оценки степени регулярности и непредсказуемости флуктуаций в данных временных рядах.

Например, есть две серии данных:

- 1) серия 1: (10, 20, 10, 20, 10, 20, 10, 20, 10, 20, 10, 20 ...), в которой чередуются 10 и 20.
- 2) серия 2: (10, 10, 20, 10, 20, 20, 20, 10, 10, 20, 10, 20, 20 ...), которая имеет значение 10 или 20, выбранное случайным образом, каждый с вероятностью 1/2.

Статистика моментов, такая как среднее значение и дисперсия, не будет различать эти две серии. Статистика ранговых порядков также не будет различать эти ряды. Тем не менее серия 1 «совершенно регулярна»; знание того, что один термин имеет значение 20, позволяет с уверенностью предсказать, что следующий термин будет иметь значение 10. Серия 2 оценивается случайным образом; знание того, что один термин имеет значение 20, не дает представления о том, какое значение будет иметь следующий термин.

Регулярность первоначально измерялась точной статистикой регулярности, которая в основном концентрировалась на различных показателях энтропии. Тем не менее для точного расчета энтропии требуются огромные объемы данных, и на результаты будет сильно влиять системный шум, поэтому применять эти методы к экспериментальным данным нецелесообразно. *ApEn* был разработан Стивом М. Пинкусом для обработки этих ограничений путем изменения точной статистики регулярности, энтропии Колмогорова — Синяя. Параметр *ApEn* изначально был разработан для анализа медицинских данных, таких как частота сердечных сокращений, а затем его приложение было распространено в области финансов, психологии и инженерии человеческого фактора. Ниже представлена реализация функции нахождения *ApEn*. Здесь m — размерность вложения (как правило, используется $m=2$), r — параметр близости (как правило, используется значения $r=0.2 \cdot SD$) [86].

```
def ApEn(data, m, r):
    def _maxdist(x_i, x_j):
        return max([abs(ua - va) for ua,
                    va in zip(x_i, x_j)])
```

```

def _phi(m):
    x = [[data[j] for j in range(i, i + m - 1 + 1)]
          for i in range(N - m + 1)]
    C = [len([1 for x_j in x if _maxdist(x_i, x_j) <=
r]) / (N - m + 1.0) for x_i in x]
    return (N - m + 1.0)**(-1) * sum(np.log(C))
N = len(U)
return abs(_phi(m+1) - _phi(m))

```

Sample Entropy

Выборочная энтропия (*SampEn*) является модификацией приближительной энтропии (*ApEn*), используемой для оценки сложности физиологических сигналов временных рядов, диагностики болезненных состояний. *SampEn* имеет два преимущества перед *ApEn*: независимость от длины данных и относительно безаварийную реализацию. Кроме того, есть небольшая вычислительная разница: в *ApEn* сравнение между вектором шаблона и остальными векторами также включает сравнение с самим собой. Следовательно, всегда можно взять логарифм вероятностей. Поскольку сравнения шаблонов с самими собой имеют более низкие значения *ApEn*, сигналы интерпретируются как более регулярные, чем они есть на самом деле. Эти совпадения не включены в *SampEn*. Ниже представлена реализация функции нахождения *SampEn* [86].

```

def SampEn(data, m, r):
    def _maxdist(x_i, x_j):
        result = max([abs(ua - va) for ua,
va in zip(x_i, x_j)])
        return result

    def _phi(m):
        x = [[data[j] for j in range(i, i + m - 1 + 1)]
              for i in range(N - m + 1)]
        C = 0
        for i in range(len(x)):
            for j in range(len(x)):

```

```

        if i == j:
            continue
        C += (_maxdist(x[i], x[j]) <= r)
    return C
N = len(data)
return -np.log(_phi(m + 1) / _phi(m))

```

График Пуанкаре

График Пуанкаре, названный в честь Анри Пуанкаре, представляет собой тип рекуррентного графика, используемого для количественной оценки самоподобия в процессах, обычно периодических функций. Он также известен как карта возврата. Графики Пуанкаре могут использоваться, чтобы отличить хаос от случайности, встраивая набор данных в многомерное пространство состояний.

В контексте сигнала ВСР график Пуанкаре представляет собой график $R-R(n)$ на оси x в зависимости от $R-R(n+1)$ (последующий интервал $R-R$) на оси y , то есть каждый берет последовательность интервалов и строит график каждого интервала относительно следующего интервала. График повторения используется в качестве стандартного метода визуализации для обнаружения присутствия колебаний в нелинейных динамических системах. В контексте электрокардиографии частота здорового сердца обычно строго контролируется регулирующими механизмами организма (в частности, вегетативной нервной системой). Несколько исследовательских работ демонстрируют потенциал основанных на сигнале ЭКГ графиков Пуанкаре в выявлении связанных с сердцем заболеваний или аномалий.

Количественный анализ графика влечет за собой подгонку эллипса к графику, центр которого совпадает с центральной точкой меток, и сравнение точек из двух линий, проходящих через центральную точку данных, продольную и поперечную. Линия, описывающая наклон продольной оси, была определена как длинная ось, а поперечный наклон, который перпендикулярен в направлении к длинной оси, был определен как короткая ось. Длина поперечной линии определяется как $SD1$ данных графика в перпендикулярном направлении. Длина продольной линии определяется как $SD2$ данных графика. Индексы Пуанкаре ($SD1$, $SD2$, $SD1/SD2$) могут быть рассчитаны с использованием следующих команд:

```

NN_plus=NN[0:-1]
NN_minus=NN[1:]
SD1=np.sqrt(0.5)*np.std(NN_plus - NN_minus))
SD2=np.sqrt(2*(np.std(NN)**2)-0.5*(np.std(NN_plus -
NN_minus)**2)))

```

Построение странного аттрактора

Одним из самых простых и наглядных способов оценки хаотического поведения системы является построение аттрактора фазовой траектории движения системы. Данный метод применяется для анализа статистических и фрактальных свойств аттракторов фазовых траекторий.

В фазовом пространстве колебательной системы предельное множество точек, притягивающее фазовые траектории, называется аттрактором (от глагола *to attract* — притягивать). Аттракторы могут быть простыми, регулярными, с установившимся режимом и так называемые «странные» аттракторы с непериодическими колебаниями. Странный аттрактор в отличие от регулярного не является многообразием (т.е. не является кривой или поверхностью), его геометрическое устройство очень сложно, а его структура фрактальна. Такие аттракторы обладают геометрической (масштабной) инвариантностью, или, другими словами, скейлинговой структурой.

Для таких сложных систем, как живые организмы, по измеренному ВР одной наблюдаемой динамической переменной можно сконструировать странный аттрактор. Согласно теореме Такенса, основные свойства аттрактора будут такими же, как у исследуемого объекта, и на основе подобия можно определить его характеристики и попытаться построить математическую модель исходной системы [87]. Моделирование не является целью данного учебного пособия, его задача — показать возможности методов нелинейной динамики оценивать информационно значимые параметры сложных систем.

В общем случае фазовое пространство является N -мерным. Однако для практических случаев можно считать пространство отображения трехмерным, т.е. $N = 3$.

```

fig=plt.figure()
fig.set_figheight(20)
fig.set_figwidth(20)
plt.plot(NN[0:-2], np.diff(NN)[0:-1], np.diff
(np.diff(NN)), 'k-*', alpha=0.2, linewidth=0.5,)

```

Практическое задание к п. 4.7

Выполните анализ нелинейных параметров биосигнала, импортированного при выполнении практического задания к главе 3:

- 1) выполните расчет нелинейных параметров биосигнала;
- 2) визуализируйте график Пуанкаре и аттрактор;
- 3) сохраните полученные результаты и изображения.

Контрольные вопросы к главе 4

1. Перечислите этапы предварительной обработки биосигналов.
2. Перечислите статистические параметры. Укажите особые статистические параметры для сигналов ВСР.
3. Сформулируйте алгоритм построения гистограмм. Перечислите виды гистограмм. Укажите особенности построения гистограмм для сигналов ВСР.
4. Опишите Фурье-преобразование биосигналов. Укажите характерные частоты для разных биосигналов.
5. Охарактеризуйте принципиальное отличие преобразования Фурье от вейвлет-преобразования.
6. Охарактеризуйте отличие аппроксимированной энтропии от выборочной энтропии.

Глава 5.

Применение методов машинного обучения для анализа биомедицинских сигналов*

Конечной целью анализа биомедицинских сигналов является распознавание или классификация сигнала с целью отнесения его к одной из нескольких известных категорий и получение диагностического решения, относящегося к патологическому состоянию пациента. Нужно учитывать, что биомедицинские сигналы являются лишь одним из источников информации для постановки диагноза, поэтому результат классификации сигнала может помочь в процедуре диагностики, но не является единственным фактором. Однако классификация образов, основанная на анализе сигнала, действительно является важным аспектом анализа биомедицинских сигналов.

Вид биомедицинского сигнала слишком сложен, чтобы был возможен его тщательный визуальный анализ. Трудно осуществить ручную текущий мониторный контроль ЭКГ тяжелых кардиологических больных с нарушениями сердечного ритма. Разработка алгоритмов для анализа биомедицинских сигналов, однако, является непростой задачей. Изменчивость и разнообразие признаков в биомедицинских сигналах поразительны. Тенденции последнего времени направлены на распознавание физиологических систем через анализ сигналов. Это диагностика, поддерживаемая компьютером, которая тесно связана с проблемами обработки сигналов, распознавания образов и автоматического обучения компьютерных систем.

При разработке алгоритмов анализа биомедицинских сигналов используются такие методы, как адаптивная фильтрация, сегментация

* Авторы главы 5: А. П. Немирко, Л. А. Манило (Санкт-Петербургский государственный электротехнический университет, г. Санкт-Петербург)

и аппроксимация, статистическое обнаружение, корреляционное распознавание, синхронное накопление, динамический кластерный анализ формы биосигналов, структурное распознавание фрагментов. В главе рассматриваются методы машинного обучения по прецедентам или индуктивного обучения, которое основано на выявлении эмпирических закономерностей в данных. Многие методы машинного обучения являются альтернативой классическим статистическим подходам. Они тесно связаны с интеллектуальным анализом данных (*data mining*).

Достаточно распространенной библиотекой для задач машинного обучения в *Python* является библиотека `scikit-learn`. Эта библиотека содержит простые и эффективные инструменты для анализа данных, основанного на библиотеках `NumPy`, `SciPy`, и `matplotlib`.

5.1. Методы классификации

Методы классификации применяются для задач анализа данных в случае, когда важно определить степень сходства или различия классов. Часто необходимо создать алгоритм автоматического определения (распознавания) принадлежности объектов к определенным классам. Классификация относится к автоматизации интеллектуальных задач принятия решений и опирается на теорию вероятностей, теорию матриц, математическую логику и математическую лингвистику. Методам классификации посвящена обширная литература ([88–92] и др.).

Можно привести следующие примеры применения методов классификации в различных областях человеческой деятельности:

- диагностика заболеваний [67; 93];
- распознавание и прогнозирование угрожающих жизни состояний больного, психофизиологических состояний человека по комплексу электрофизиологических показателей [94];
- автоматический анализ и интерпретация биомедицинских сигналов и изображений [67; 95];
- другие задачи: машинное зрение; распознавание речи; идентификация человека по голосу, отпечаткам пальцев, изображению глаза; чтение печатного и рукописного текста, топографических карт и т. п.

Представление данных в многомерном пространстве признаков

Распознавание или классификацию образов можно определить как отнесение входных данных к какому-либо из идентифицируемых классов посредством дифференциации выделенных существенных признаков.

Допустим, мы имеем m классов объектов $\omega_1, \omega_2, \dots, \omega_m$. Каждый поступающий объект должен быть отнесен к одному из этих m классов и каждый объект описывается набором из n измерений (действительных чисел) $x_1, x_2, \dots, x_n$, которые называются признаками. Такой набор может быть представлен точкой в n -мерном евклидовом пространстве E^n или, иначе говоря, вектором признаков $\mathbf{X} = (x_1, x_2, \dots, x_n)$ в n -мерном признаковом пространстве, где $x_1, x_2, \dots, x_n$ — декартовы координаты конца вектора $\mathbf{X}$. Например, при диагностике состояний больного мы имеем два класса: ω_1 — А-состояние, ω_2 — В-состояние. Каждый поступающий больной описывается тремя признаками (показателями), полученными по результатам его объективного обследования: x_1 — частота пульса, x_2 — частота дыхания, x_3 — среднее артериальное давление.

В этом случае состояние одного больного, описываемое точкой (вектором) $\mathbf{X} = (x_1, x_2, x_3)$ в трехмерном пространстве, должно быть отнесено к одному из двух классов: ω_1 или ω_2 (рис. 5.1).

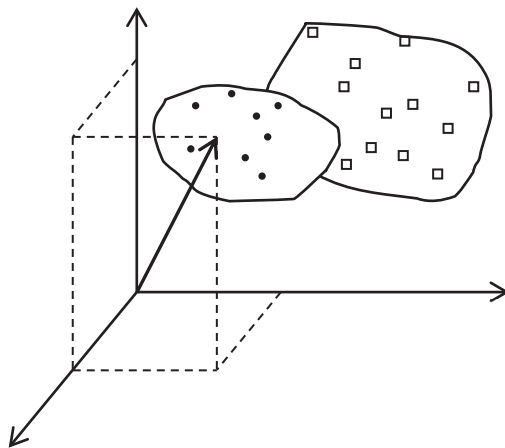


Рис. 5.1. Описание классов в признаковом пространстве

Разделяющие поверхности в признаковом пространстве

Таким образом, классификатор объектов является устройством, которое отображает точки из E^n в множество классов $\{\omega_1, \omega_2, \dots, \omega_m\}$ (рис. 5.2). Обозначим через R_i множество точек из E^n , которое отображается в класс ω_i . Тогда множеству классов $\{\omega_1, \omega_2, \dots, \omega_m\}$ будет соответствовать множество областей из E^n $\{R_1, R_2, \dots, R_m\}$. Все области отделены друг от друга поверхностями, которые мы будем называть разделяющими.

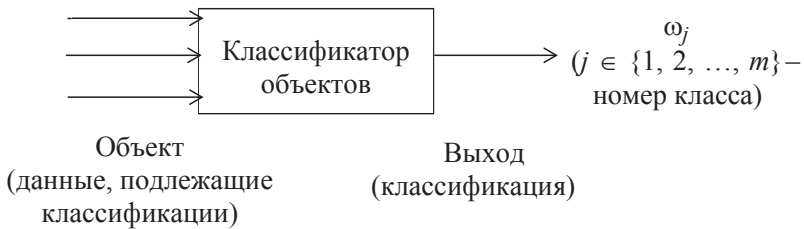


Рис. 5.2. Функция классификатора объектов

Разделяющие поверхности любого классификатора объектов можно полностью определить m скалярными функциями: $D_1(\mathbf{X}), \dots, D_m(\mathbf{X})$, называемыми дискриминантными функциями (ДФ). Они выбираются так, чтобы для всех $\mathbf{X} \in R_i$ выполнялось $D_i(\mathbf{X}) > D_j(\mathbf{X})$ при $i, j = 1, \dots, m$; $i \neq j$. Мы также предполагаем, что ДФ непрерывны на разделяющих поверхностях. Тогда поверхность, разделяющая смежные области R_i и R_j , определяется уравнением

$$D_i(\mathbf{X}) - D_j(\mathbf{X}) = 0. \quad (5.1)$$

На рис. 5.3 показаны три ДФ и области решений, порождаемые ими на плоскости x_1, x_2 .

Введение понятия дискриминантных функций преобразует модель классификатора объектов к виду рис. 5.4, где дискриминаторами названы блоки, вычисляющие дискриминантные функции. Эти функции могут принадлежать к разным видам (линейные, кусочно-линейные, нелинейные, квадратичные). При решении задач распознавания образов существует проблема выбора вида ДФ.

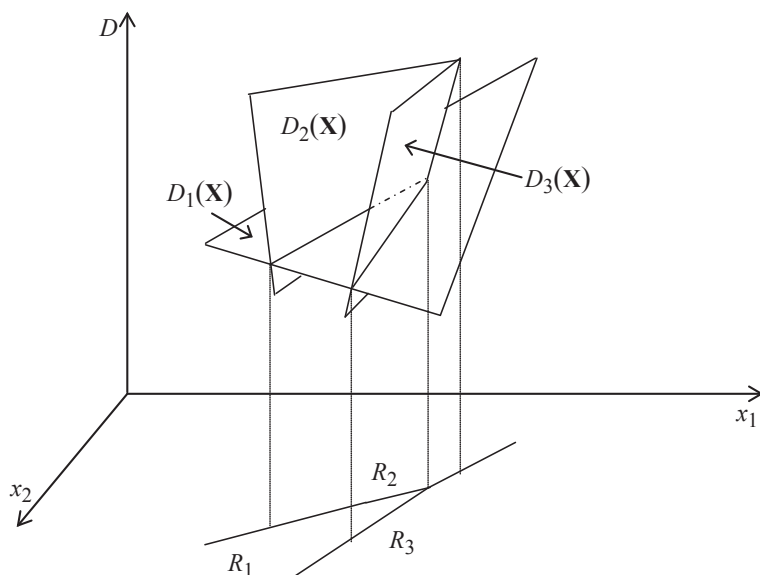


Рис. 5.3. Дискриминантные функции

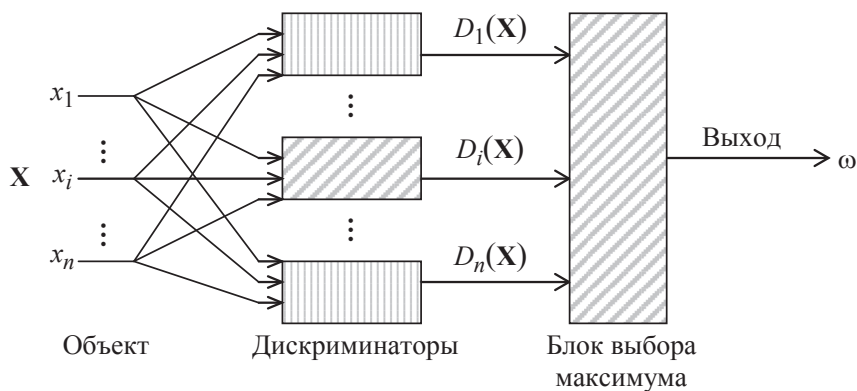


Рис. 5.4. Распознавание с помощью дискриминантных функций

Линейные дискриминантные функции

К простейшим дискриминантным функциям относятся линейные (ЛДФ), определяемые выражением

$$D_i(\mathbf{X}) = \sum_{k=1}^n w_{ik} x_k + w_{i(n+1)}, \quad i = 1, \dots, m.$$

Для $i = 1$ ЛДФ имеет вид

$$D_1(\mathbf{X}) = w_{11}x_1 + w_{12}x_2 + w_{13}x_3 + \dots + w_{1(n+1)}.$$

В этом случае поверхность, разделяющая смежные области R_i и R_j , в E^n определяется из (5.1)

$$D_i(\mathbf{X}) - D_j(\mathbf{X}) = \sum_{k=1}^n w_k x_k + w_{n+1} = 0, \quad (5.2)$$

где $w_k = w_{ik} - w_{jk}$, $w_{n+1} = w_{i(n+1)} - w_{j(n+1)}$.

Полученное уравнение — это уравнение гиперплоскости в пространстве E^n .

Если $m = 2$, то уравнение (5.2) превращается в следующее:

$$D(\mathbf{X}) = D_1(\mathbf{X}) - D_2(\mathbf{X}) = w_1x_1 + w_2x_2 + w_3x_3 + \dots + w_{n+1} = 0$$

и по знаку новой дискриминантной функции $D(\mathbf{X})$ мы можем отнести входной объект к одному из двух классов — ω_1 или ω_2 .

Если $D(\mathbf{X}) > 0$, то R_1 и входной объект относятся к классу ω_1 .

Если $D(\mathbf{X}) < 0$, то R_2 и входной объект относятся к классу ω_2 .

Введя пороговый логический блок (ПЛБ), классификатор для этого случая можно изобразить, как показано на рис. 5.5.

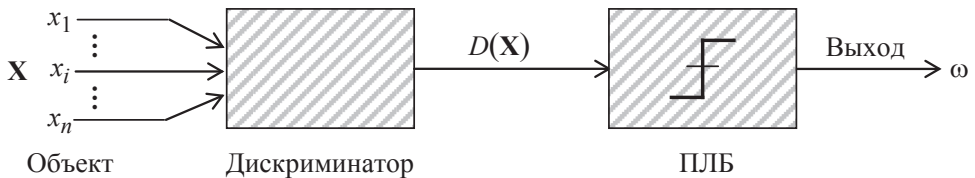


Рис. 5.5. Классификатор на два класса

Выше приведенное выражение для $D(\mathbf{X})$ можно также записать в виде

$$D(\mathbf{X}) = \mathbf{W}^T \mathbf{X} - \alpha = 0,$$

где $\mathbf{W}$ — весовой вектор; α — пороговая величина. Удобно сделать вектор $\mathbf{W}$ единичным, разделив все компоненты на норму этого вектора. В этом случае так как $\mathbf{W}^T \mathbf{X} = |\mathbf{X}| \cos \alpha$ (рис. 5.6), то процедура распознавания заключается в вычислении проекции вектора $\mathbf{X}$ на направление весового вектора $\mathbf{W}$ и сравнении полученной величины с порогом α . Также ясно, что разделяющая гиперплоскость проходит перпендикулярно к $\mathbf{W}$ на расстоянии α от начала координат.

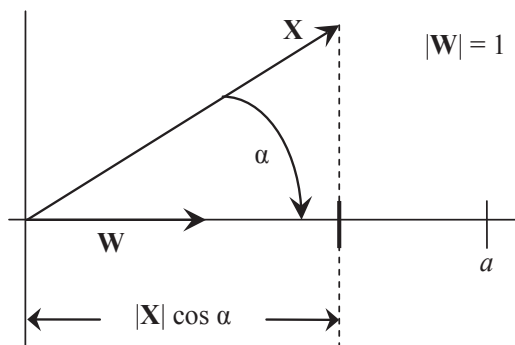


Рис. 5.6. Проекция вектора признаков на весовой вектор

Существует ряд методов определения $\mathbf{W}$ и a , основанных как на критериях теории статистических решений, так и на максимизации определенного некоторым образом расстояния между классами.

Классификатор ЭЭГ

Примером линейного классификатора является классификатор ЭЭГ (рис. 5.7). На вход устройства подается сигнал ЭЭГ. В устройстве вычисляется шесть признаков (параметров):

- x_1 — число пересечений нулевой линии;
- x_2 — средняя амплитуда исходной ЭЭГ;
- x_3 — число максимумов и минимумов;
- x_4 — средняя амплитуда 1-й производной;
- x_5 — число точек перегиба;
- x_6 — средняя амплитуда 2-й производной.

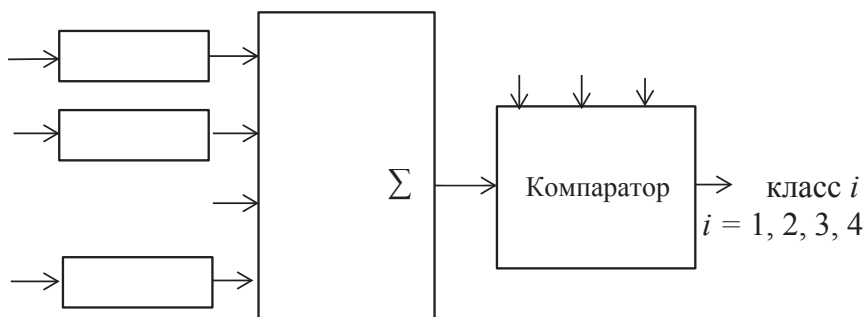


Рис. 5.7. Классификатор ЭЭГ

Эти параметры (признаки) суммируются с весами. Тогда их сумма равна

$$S = \sum_{i=1}^6 w_i x_i,$$

где w_i — веса признаков. Так как классификатор классифицирует ЭЭГ на четыре класса, вводятся три порога: w_7 , w_8 , w_9 . Тогда классификация на четыре класса производится путем сравнения с этими порогами по следующему правилу:

если $\sum_{i=1}^6 w_i x_i < w_7$, то ЭЭГ относится к классу 1, иначе

если $w_7 \leq \sum_{i=1}^6 w_i x_i < w_8$, то ЭЭГ относится к классу 2, иначе

если $w_8 \leq \sum_{i=1}^6 w_i x_i < w_9$, то ЭЭГ относится к классу 3, иначе

если $\sum_{i=1}^6 w_i x_i > w_9$, то ЭЭГ относится к классу 4.

Данный классификатор используется для определения глубины наркоза, стадий сна, изучения биологических обратных связей, влияния лекарственных препаратов на ЭЭГ и других целей.

Классификатор по минимуму расстояния

Пусть в n -мерном признаковом пространстве мы имеем m эталонных векторов $\mathbf{P}_1, \mathbf{P}_2, \dots, \mathbf{P}_m$, каждый из которых является эталоном (шаблоном) своего класса (эталонным вектором), как изображено на рис. 5.8.

Вектор $\mathbf{P}_i = (p_{i1}, p_{i2}, \dots, p_{in})$ соответствует классу ω_i . Классификатор по минимуму расстояния относит входной вектор $\mathbf{X}$ к тому классу ω_i , для которого расстояние $|\mathbf{X} - \mathbf{P}_i|$ между $\mathbf{X}$ и эталонным вектором $\mathbf{P}_i$ минимально. Вместо расстояния $|\mathbf{X} - \mathbf{P}_i|$ можно взять его квадрат $|\mathbf{X} - \mathbf{P}_i|^2$. Так как $|\mathbf{X} - \mathbf{P}_i| = \sqrt{(\mathbf{X} - \mathbf{P}_i)^T (\mathbf{X} - \mathbf{P}_i)}$, то $|\mathbf{X} - \mathbf{P}_i|^2 = (\mathbf{X} - \mathbf{P}_i)^T (\mathbf{X} - \mathbf{P}_i)$ и на основании свойства дистрибутивности скалярных произведений векторов имеем

$$|\mathbf{X} - \mathbf{P}_i|^2 = \mathbf{X}^T \mathbf{X} - \mathbf{X}^T \mathbf{P}_i - \mathbf{P}_i^T \mathbf{X} + \mathbf{P}_i^T \mathbf{P}_i \quad (5.3)$$

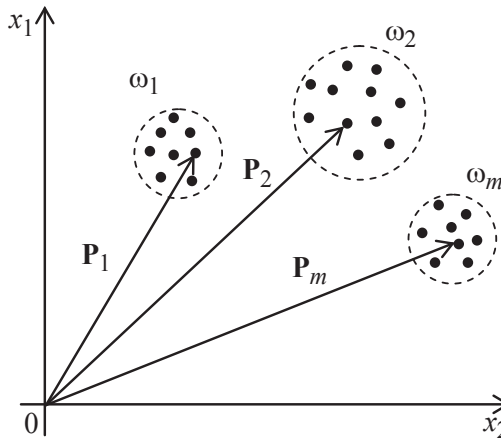


Рис. 5.8. Расположение эталонов в многомерном пространстве признаков

Произведение $\mathbf{X}^T \mathbf{X} = |\mathbf{X}|$ на расстояние не влияет, а на основании свойства коммутативности скалярных произведений векторов справедливо равенство $\mathbf{X}^T \mathbf{P}_i = \mathbf{P}_i^T \mathbf{X}$, так как

$$\sum_{j=1}^n x_j p_{ij} = \sum_{j=1}^n p_{ij} x_j.$$

Тогда, исключив $\mathbf{X}^T \mathbf{X}$ из выражения (5.3), поменяв знаки и поделив все члены на 2, получим дискриминантную функцию $D_i(\mathbf{X})$ для класса ω_i . Она имеет вид

$$D_i(\mathbf{X}) = \mathbf{X}^T \mathbf{P}_i - \frac{1}{2} \mathbf{P}_i^T \mathbf{P}_i. \quad (5.4)$$

Как и ранее, входной вектор $\mathbf{X}$ относится к классу ω_i , если

$$D_i(\mathbf{X}) > D_j(\mathbf{X}), i, j = 1, \dots, m, i \neq j.$$

Выражение (5.4) также можно написать в виде

$$D_i(\mathbf{X}) = w_{i1}x_1 + w_{i2}x_2 + w_{i3}x_3 + \dots + w_{i(n+1)}, i = 1, \dots, m,$$

где $w_{ij} = p_{ij}$, $i = 1, \dots, m, j = 1, \dots, n$; $w_{i(n+1)} = -\frac{1}{2}(p_{i1}^2 + p_{i2}^2 + \dots + p_{in}^2)$, $i = 1, \dots, m$.

Таким образом, видно, что классификатор по минимуму расстояния является линейным классификатором. Такой классификатор целесообразно применять в случаях, когда каждый класс имеет свой эталонный объект $\mathbf{P}_i$, вокруг которого группируются остальные объекты этого класса.

При $m = 2$ (два класса)

$$D_1(\mathbf{X}) = \mathbf{X}^T \mathbf{P}_1 - \frac{1}{2} \mathbf{P}_1^T \mathbf{P}_1, \quad D_2(\mathbf{X}) = \mathbf{X}^T \mathbf{P}_2 - \frac{1}{2} \mathbf{P}_2^T \mathbf{P}_2,$$

$$D^0(\mathbf{X}) = D_1(\mathbf{X}) - D_2(\mathbf{X}) = \mathbf{X}^T \mathbf{P}_1 - \mathbf{X}^T \mathbf{P}_2 - \frac{1}{2} (\mathbf{P}_1^T \mathbf{P}_1 - \mathbf{P}_2^T \mathbf{P}_2).$$

Добавим к выражению в скобках член $\mathbf{P}_1^T \mathbf{P}_2$ и вычтем равный ему $\mathbf{P}_2^T \mathbf{P}_1$. Тогда

$$D^0(\mathbf{X}) = \mathbf{X}^T (\mathbf{P}_1 - \mathbf{P}_2) - \frac{1}{2} (\mathbf{P}_1^T \mathbf{P}_1 - \mathbf{P}_1^T \mathbf{P}_2 + \mathbf{P}_2^T \mathbf{P}_1 - \mathbf{P}_2^T \mathbf{P}_2),$$

$$D^0(\mathbf{X}) = \mathbf{X}^T (\mathbf{P}_1 - \mathbf{P}_2) - \frac{1}{2} (\mathbf{P}_1 + \mathbf{P}_2)^T (\mathbf{P}_1 - \mathbf{P}_2).$$

Разделим все члены последнего выражения на $|\mathbf{P}_1 - \mathbf{P}_2|$. Тогда

$$D(\mathbf{X}) = \mathbf{X}^T \frac{\mathbf{P}_1 - \mathbf{P}_2}{|\mathbf{P}_1 - \mathbf{P}_2|} - \frac{1}{2} \frac{(\mathbf{P}_1 + \mathbf{P}_2)^T (\mathbf{P}_1 - \mathbf{P}_2)}{|\mathbf{P}_1 - \mathbf{P}_2|},$$

$$D(\mathbf{X}) = \mathbf{W}^T \mathbf{X} - \alpha = 0, \quad \mathbf{W} = \frac{\mathbf{P}_1 - \mathbf{P}_2}{|\mathbf{P}_1 - \mathbf{P}_2|}, \quad \alpha = -\frac{1}{2} (\mathbf{P}_1 + \mathbf{P}_2)^T \mathbf{W}$$

На рис. 5.9 показана разделяющая (линейная) граница $D(\mathbf{X}) = 0$ между двумя классами, проведенная для классификации по минимуму расстояния до эталонов классов.

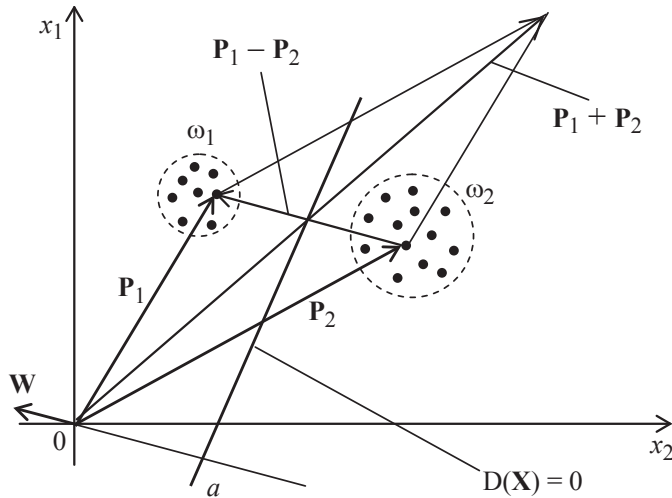


Рис. 5.9. Классификация по минимуму расстояния до эталонов двух классов

5.2. Метод главных компонент

Во многих задачах обработки и анализа биомедицинских сигналов и данных исходное число p рассматриваемых, т. е. измеряемых на исследуемых объектах признаков довольно велико. Для визуализации картины, простоты интерпретации и упрощения расчетов очень часто необходимо представить каждое из наблюдений в виде набора чисел, состоящего из *существенно меньшего*, чем p , числа признаков. При этом оставшиеся признаки могут либо выбираться из числа исходных, либо определяться по какому-либо правилу по совокупности исходных признаков, например, как линейные комбинации последних.

Главные компоненты представляют собой новое множество исследуемых признаков $y^{(1)}, y^{(2)}, \dots, y^{(p)}$, каждый из которых получен в результате некоторой линейной комбинации непосредственно измеренных на объектах исходных признаков $x^{(1)}, x^{(2)}, \dots, x^{(p)}$. Полученные в результате такого преобразования новые признаки $y^{(1)}, y^{(2)}, \dots, y^{(p)}$ могут обладать рядом удобных статистических свойств, например, быть упорядоченными по степени рассеяния в изучаемой совокупности объектов: первый признак обладает наибольшей степенью рассеяния, т. е. наибольшей дисперсией, второй — меньшей, и т. д.

Рассмотрим метод вычисления главных компонент [96]. Будем предполагать, что исследуемые наблюдения $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n$ извлечены из некоторой p -мерной генеральной совокупности, определяемой соответствующей вероятностной мерой. Из всех характеристик исследуемой совокупности данных существенное значение имеет ковариационная матрица $\Sigma = [\sigma_{ij}]$, где

$$\sigma_{ij} = M(x^{(i)} - a^{(i)})(x^{(j)} - a^{(j)}), i, j = 1, 2, \dots, p \quad (5.5)$$

Здесь $a^{(i)}$ — компоненты вектора $\mathbf{a}$ средних значений признаков $x^{(i)}$. Будем считать, что в (5.5) $\mathbf{a} = 0$. Этого всегда можно добиться, рассматривая в качестве исходных признаков $x^{(1)}, x^{(2)}, \dots, x^{(p)}$ не сами измерения, а их отклонения от выборочных средних значений.

Первой главной компонентой исследуемой совокупности наблюдений называется такая нормированная линейная комбинация p исходных признаков $x^{(1)}, x^{(2)}, \dots, x^{(p)}$:

$$y^{(1)} = l_{11}x^{(1)} + l_{12}x^{(2)} + \dots + l_{1p}x^{(p)} = \mathbf{L}_1^T \mathbf{X}$$

$$(\mathbf{L}_1^T = [l_{11}, l_{12}, \dots, l_{1p}], l_{11}^2 + l_{12}^2 + \dots + l_{1p}^2 = 1),$$

которая среди всех прочих нормированных линейных комбинаций $x^{(1)}, x^{(2)}, \dots, x^{(p)}$ обладает *наибольшей* дисперсией.

Вообще, i -й *главной компонентой* исследуемой генеральной совокупности ($i = 2, 3, \dots, p$) называют такую нормированную линейную комбинацию p исходных признаков

$$y^{(i)} = l_{i1}x^{(1)} + l_{i2}x^{(2)} + \dots + l_{ip}x^{(p)} = \mathbf{L}_i^T \mathbf{X},$$

которая среди всех прочих линейных нормированных $l_{i1}^2 + l_{i2}^2 + \dots + l_{ip}^2 = 1$ комбинаций, некоррелированных со всеми предшествующими главными компонентами $y^{(1)}, \dots, y^{(i-1)}$, (т.е. $\text{cov}(y^{(i)}, y^{(j)}) = M(y^{(i)}y^{(j)}) = 0$ для $j < i$), обладает *наибольшей* дисперсией.

Из определения следует, что, во-первых, главные компоненты $y^{(1)}, y^{(2)}, \dots, y^{(p)}$ пронумерованы в порядке убывания их дисперсий, т.е. $Dy^{(1)} \geq Dy^{(2)} \geq \dots \geq Dy^{(p)}$, где

$$Dy^{(i)} = M(\mathbf{L}_i^T \mathbf{X})^2 = M(\mathbf{L}_i^T \mathbf{X} \mathbf{X}^T \mathbf{L}_i) = \mathbf{L}_i^T \Sigma^T \mathbf{L}_i \quad (5.6)$$

и, во-вторых, вектор $\mathbf{L}_i$, определяющий переход от $x^{(1)}, x^{(2)}, \dots, x^{(p)}$ к $y^{(i)}$ является i -м *собственным вектором* ковариационной матрицы Σ , т.е. его компоненты $l_{i1}, l_{i2}, \dots, l_{ip}$ определяются как нормированное ($\sum_{j=1}^p l_{ij}^2 = 1$) решение системы уравнений

$$(\Sigma - \lambda_i \mathbf{I}) \mathbf{L}_i = 0. \quad (5.7)$$

Эта система линейных однородных уравнений имеет ненулевое решение только в том случае, когда ее определитель равен нулю, т.е.

$$(\Sigma - \lambda_i \mathbf{I}) \mathbf{L}_i = 0, \quad (5.8)$$

где $|\cdot|$ означает определитель матрицы, λ — неизвестное число, $\mathbf{I}$ — единичная матрица.

Выражение (5.8), в котором определитель матрицы $(\Sigma - \lambda \mathbf{I})$ является многочленом степени p относительно λ называют *характеристическим уравнением*, а его корни $\lambda_i, i = 1, 2, \dots, p$ *собственными значениями* матрицы Σ .

Каждому собственному значению λ_i , определяемому из (5.8), соответствует свой собственный вектор $\mathbf{L}_i$ матрицы Σ . Составляющие этого вектора $l_{i1}, l_{i2}, \dots, l_{ip}$ для конкретного значения λ_i могут быть найдены путем решения системы уравнений (5.7).

Из сопоставления (5.6)–(5.8) следует, что

$$Dy^{(i)} = \mathbf{L}_i^T \Sigma \mathbf{L}_i = \mathbf{L}_i^T \lambda_i \mathbf{I} \mathbf{L}_i = \lambda_i.$$

Таким образом, ковариационная матрица Σ_Y главных компонент $y^{(1)}, y^{(2)}, \dots, y^{(p)}$ будет иметь вид диагональной матрицы

$$\Sigma_Y = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \lambda_p \end{bmatrix}.$$

Учитывая, что преобразование, с помощью которого осуществляется переход от исходных компонент $\mathbf{X}$ к главным компонентам $\mathbf{Y}$ ($\mathbf{Y} = \mathbf{L}\mathbf{X}$), является *ортogonalным*, исходные переменные $x^{(1)}, x^{(2)}, \dots, x^{(p)}$ можно выразить через главные компоненты

$$x^{(i)} = l_{i1}y^{(1)} + l_{i2}y^{(2)} + \dots + l_{ip}y^{(p)}$$

(в матричной записи $\mathbf{X} = \mathbf{L}^T \mathbf{Y}$).

Несмотря на то, что для p исходных признаков получено p главных компонент, вклад большей их части в суммарную дисперсию оказывается небольшим. Поэтому, исключив из рассмотрения те компоненты, вклад которых мал, можно существенно сократить размерность исследуемого признакового пространства.

Ортogonalное преобразование $\mathbf{Y} = \mathbf{L}\mathbf{X}$ оставляет инвариантной обобщенную дисперсию, а также сумму дисперсий компонент исходного вектора $\mathbf{X}$ и, как следствие, суммарная дисперсия ($Dy^{(1)} + Dy^{(2)} + \dots + Dy^{(p)}$) главных компонент, равная сумме собственных чисел $(\lambda_1 + \lambda_2 + \dots + \lambda_p)$, также равна сумме дисперсий ($Dx^{(1)} + Dx^{(2)} + \dots + Dx^{(p)}$) исходных признаков [96]. На этом основании принимается решение о том, сколько последних главных компонент можно исключить из последующего рассмотрения. Для этого проводится анализ изменения относительной доли дисперсии

$$q(p') = \frac{Dy^{(1)} + Dy^{(2)} + \dots + Dy^{(p')}}{Dy^{(1)} + Dy^{(2)} + \dots + Dy^{(p)}} = \frac{\lambda_1 + \lambda_2 + \dots + \lambda_{p'}}{\lambda_1 + \lambda_2 + \dots + \lambda_p} \quad (5.9)$$

($1 \leq p' \leq p$), вносимой первыми p' главными компонентами, и определяется число компонент, которое целесообразно сохранить. Так, для примера кривой $q(p')$, изображенной на рис. 5.10, очевидно возможно сокращение размерности пространства с $p = 10$ до $p' = 3$, так как вклад семи последних главных компонент в суммарное рассеяние составляет не более 10 %.

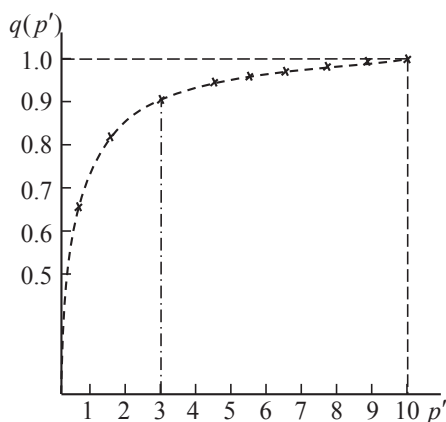


Рис. 5.10. Зависимость относительной доли суммарной дисперсии исследуемых признаков $q(p')$ от числа p' первых главных компонент (случай $p = 10$)

Анализируя лишь $p' = 3$ первых, т. е. наиболее весомых главных компонент, можно объяснить основную долю, а именно 90 % суммарной дисперсии и выявить особенности рассеяния или группировки наблюдений.

Такой анализ главных компонент целесообразен, если величины $x^{(1)}, x^{(2)}, \dots, x^{(p)}$, являющиеся составляющими вектора $\mathbf{X}$, имеют одинаковую размерность. В случае если признаки $x^{(1)}, x^{(2)}, \dots, x^{(p)}$ измеряются в различных физических единицах, на этапе предобработки необходимо перейти к безразмерным признакам $x_v^{*(i)}$, например, с помощью нормирующего преобразования вида

$$x_v^{*(i)} = \frac{x_v^{(i)}}{\sqrt{\sigma_{ii}}}, \quad (i = 1, 2, \dots, p, v = 1, 2, \dots, n),$$

где σ_{ii} соответствует (5.5), а затем вычислять главные компоненты относительно этих признаков $x^{*(1)}$, $x^{*(2)}$, ..., $x^{*(p)}$ и их ковариационной матрицы.

На рис. 5.11 приведено изображение множества ритмограмм в пространстве двух первых главных компонент для трех классов ЭКГ (МА, НР, ЧЭ). Как видно из рисунка, описание объектов параметрами энтропии позволяет получить группировку объектов, соответствующую трем классам рассматриваемых ЭКГ-сигналов. Наиболее плотная группировка объектов наблюдается в классе МА. Возможно линейное разделение классов, причем, используя энтропийные характеристики ритмограмм, можно эффективно решить задачу распознавания опасного нарушения (МА) на фоне альтернативных групп аритмий (ЧЭ и НР с выраженной дыхательной аритмией).

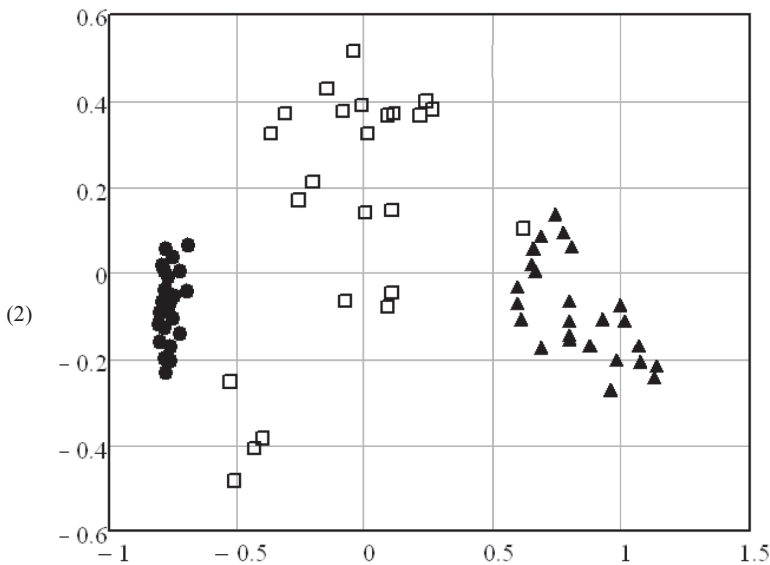


Рис. 5.11. Представление ритмограмм классов: мерцательная аритмия, ● (МА), нормальный ритм, □ (НР) и частая экстрасистолия, ▲ (ЧЭ) в пространстве двух первых главных компонент

Практическое задание к п. 5.2

Проведите анализ выборки данных, импортированных при выполнении практического задания 2 к главе 3, методом главных компонент:

- 1) совместно с преподавателем выберите выборку биомедицинских сигналов для анализа;
- 2) вычислите параметры биомедицинских сигналов, используя рекомендации, описанные в главе 4;
- 3) реализуйте алгоритм нахождения главных компонент. Сравните полученное решение с функциями библиотеки `scikit-learn`;
- 4) определите оптимальную размерность пространства параметров, используя суммарную дисперсию;
- 5) визуализируйте и сохраните полученный результат.

5.3. Линейный дискриминантный анализ

Построение решающих функций для распознавания двух классов объектов ω_1 и ω_2 по некоторому описанию $\mathbf{X} = (x_1, x_2, \dots, x_L)$ представляет собой непростую задачу. Известный байесовский подход, основанный на вычислении отношения правдоподобия, оптимален, поскольку связан с наилучшим критерием, минимизирующим вероятность ошибочного решения [88]. Однако на практике его трудно реализовать, так как для проведения вычислений необходимо знать условные плотности вероятностей, процедура оценивания которых довольно сложна. Более простым оказывается линейный классификатор, который можно построить, используя разные критерии оптимизации параметров решающей функции, в том числе критерий Фишера [89].

Для двухклассовой задачи ЛДФ, являющаяся разделяющей гиперплоскостью, имеет вид

$$D(\mathbf{X}) = \mathbf{W}^T \mathbf{X} - \alpha = 0 \text{ или } \mathbf{W}^T \mathbf{X} = \alpha, \quad (5.10)$$

где $\mathbf{W} = (w_1, w_2, \dots, w_L)$ — весовой вектор единичной длины; α — скалярная пороговая величина.

Алгоритм распознавания, позволяющий отнести неизвестный объект $\mathbf{X}$ к одной из групп, в этом случае имеет следующий вид:

- если $\mathbf{W}^T \mathbf{X} < \alpha$, то ω_1 (1-й класс),
- если $\mathbf{W}^T \mathbf{X} \geq \alpha$, то ω_2 (2-й класс).

На рис. 5.12 изображены ЛДФ и соответствующий ей алгоритм распознавания.

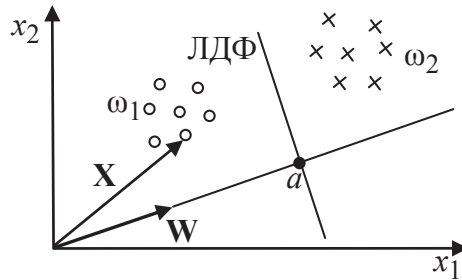


Рис. 5.12. Пример построения линейной дискриминантной функции

Для того чтобы определить составляющие вектора $\mathbf{W}$ и порог a , можно воспользоваться критерием минимизации ошибки классификации. Однако для этого потребуются довольно сложная итеративная процедура поиска параметров решающего правила. Эту задачу можно решить, применяя другой критерий оптимизации — критерий Фишера. Он позволяет найти такой вектор $\mathbf{W}$, при котором проекции точек классов на выбранное направление разделяются наилучшим образом. Порог a определяется исходя из критерия оптимальности разбиения объектов на два класса. В этом случае алгоритм поиска неизвестных параметров $\mathbf{W}$ и a можно представить следующим образом:

- 1) найти $\mathbf{W}$, используя критерий Фишера;
- 2) спроецировать точки обоих классов на прямую, определяемую положением $\mathbf{W}$;
- 3) решить одномерную задачу по поиску наилучшей величины a , например, по критерию минимума ошибок классификации.

Если допустить, что оба класса распределены по нормальному закону с одинаковыми ковариационными матрицами $\Sigma_1 = \Sigma_2 = \Sigma$, то из теории известно, что решение первого шага выше приведенного алгоритма определяется выражением

$$\mathbf{W} = \Sigma^{-1}(\mathbf{M}_1 - \mathbf{M}_2), \quad (5.11)$$

где $\mathbf{M}_1$ и $\mathbf{M}_2$ — векторы средних значений двух классов. В случае $\Sigma_1 \neq \Sigma_2$ в качестве Σ можно использовать усредненную ковариационную матрицу в виде

$$\Sigma = \frac{1}{2}(\Sigma_1 + \Sigma_2). \quad (5.12)$$

Так как после вычисления $\mathbf{W}$ по формуле (5.11) его надо нормировать для получения вектора единичной длины, т. е. разделить на $|\mathbf{W}|$, то коэффициент $\frac{1}{2}$ в (5.12) можно отбросить. Тогда $\Sigma = \Sigma_1 + \Sigma_2$.

Оценки $\mathbf{M}_1$, $\mathbf{M}_2$ и Σ_1 , Σ_2 определяются выражениями

$$\mathbf{M}_1 = \frac{1}{n_1} \sum_{i=1}^{n_1} \mathbf{X}_i^{(1)},$$

$$\mathbf{M}_2 = \frac{1}{n_2} \sum_{i=1}^{n_2} \mathbf{X}_i^{(2)},$$

$$\Sigma_j = \frac{1}{n_j - 1} \sum_{i=1}^{n_j} (\mathbf{X}_i^{(j)} - \mathbf{M}_j)(\mathbf{X}_i^{(j)} - \mathbf{M}_j)^T, j = 1, 2.$$

где $\mathbf{X}_i^{(1)}$ — i -й объект из 1-го класса; $\mathbf{X}_i^{(2)}$ — i -й объект из 2-го класса, n_1 — число объектов 1-го класса; n_2 — число объектов 2-го класса.

Если спроецировать точки объектов каждого класса на направление $\mathbf{W}$, то каждый вектор $\mathbf{X}$ превратится в соответствующий скаляр y : $y_i^{(1)} = \mathbf{W}^T \mathbf{X}_i^{(1)}$, $y_i^{(2)} = \mathbf{W}^T \mathbf{X}_i^{(2)}$ и т. д. (рис. 5.13).

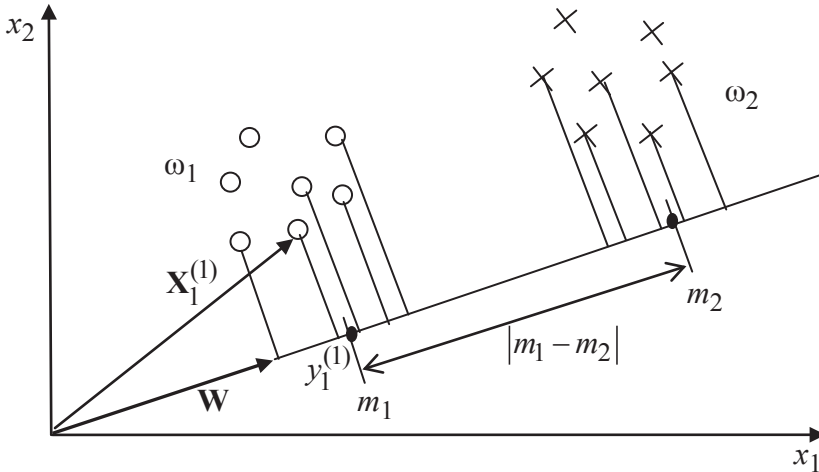


Рис. 5.13. Определение проекций объектов двух классов на прямую, продолжающую вектор $\mathbf{W}$

Тогда средние значения для выборок спроектированных точек равны

$$m_1 = \frac{1}{n_1} \sum_{i=1}^{n_1} y_i^{(1)} = \frac{1}{n_1} \sum_{i=1}^{n_1} \mathbf{w}^T \mathbf{X}_i^{(1)} = \mathbf{w}^T \frac{1}{n_1} \sum_{i=1}^{n_1} \mathbf{X}_i^{(1)} = \mathbf{w}^T \mathbf{M}_1,$$

$$m_2 = \frac{1}{n_2} \sum_{i=1}^{n_2} y_i^{(2)} = \frac{1}{n_2} \sum_{i=1}^{n_2} \mathbf{w}^T \mathbf{X}_i^{(2)} = \mathbf{w}^T \frac{1}{n_2} \sum_{i=1}^{n_2} \mathbf{X}_i^{(2)} = \mathbf{w}^T \mathbf{M}_2.$$

Очевидно, что расстояние между m_1 и m_2 будет отражать удаленность классов друг от друга после их проецирования на $\mathbf{W}$. Это расстояние, оценивающее межклассовый разброс, задается выражением

$$(m_1 - m_2)^2 = (\mathbf{w}^T \mathbf{M}_1 - \mathbf{w}^T \mathbf{M}_2)^2 =$$

$$= \mathbf{w}^T (\mathbf{M}_1 - \mathbf{M}_2)(\mathbf{M}_1 - \mathbf{M}_2)^T \mathbf{w} = \mathbf{w}^T \mathbf{S}_B \mathbf{w}, \quad (5.13)$$

где $\mathbf{S}_B = (\mathbf{M}_1 - \mathbf{M}_2)(\mathbf{M}_1 - \mathbf{M}_2)^T$ — матрица разброса между классами.

Также можно определить выборочный разброс для двух классов s_1^2 и s_2^2 в виде

$$s_1^2 = \sum_{i=1}^{n_1} (y_i^{(1)} - m_1)^2 = \sum_{i=1}^{n_1} (\mathbf{w}^T \mathbf{X}_i^{(1)} - \mathbf{w}^T \mathbf{M}_1)^2,$$

$$s_2^2 = \sum_{i=1}^{n_2} (y_i^{(2)} - m_2)^2 = \sum_{i=1}^{n_2} (\mathbf{w}^T \mathbf{X}_i^{(2)} - \mathbf{w}^T \mathbf{M}_2)^2.$$

Этот показатель, как и дисперсия, является оценкой внутриклассового рассеяния — чем он больше, тем и рассеяние данных больше. По s_1^2 и s_2^2 можно определить «средний» внутриклассовый разброс в виде $s_1^2 + s_2^2$

Для определения s_1^2 и s_2^2 через статистические параметры выборок классов определим матрицу разброса $\mathbf{S}_j$ для j -го класса (аналог корреляционной матрицы) в виде

$$\mathbf{S}_j = \sum_{i=1}^{n_j} (\mathbf{X}_i^{(j)} - \mathbf{M}_j)(\mathbf{X}_i^{(j)} - \mathbf{M}_j)^T$$

и «усредненную» матрицу разброса для двух классов в виде $\mathbf{S}_W = \mathbf{S}_1 + \mathbf{S}_2$

Тогда

$$s_j^2 = \sum_{i=1}^{n_j} (\mathbf{w}^T \mathbf{X}_i^{(j)} - \mathbf{w}^T \mathbf{M}_j)^2 = \sum_{i=1}^{n_j} \mathbf{w}^T (\mathbf{X}_i^{(j)} - \mathbf{M}_j)(\mathbf{X}_i^{(j)} - \mathbf{M}_j)^T \mathbf{w} =$$

$$= \mathbf{w}^T \left[\sum_{i=1}^{n_j} (\mathbf{X}_i^{(j)} - \mathbf{M}_j)(\mathbf{X}_i^{(j)} - \mathbf{M}_j)^T \right] \mathbf{w} = \mathbf{w}^T \mathbf{S}_j \mathbf{w}$$

$$s_1^2 + s_2^2 = \mathbf{w}^T \mathbf{S}_1 \mathbf{w} + \mathbf{w}^T \mathbf{S}_2 \mathbf{w} = \mathbf{w}^T (\mathbf{S}_1 + \mathbf{S}_2) \mathbf{w} = \mathbf{w}^T \mathbf{S}_W \mathbf{w}. \quad (5.14)$$

Теперь можно определить *линейный дискриминант Фишера* — линейную функцию с максимальным отношением разброса между классами к «среднему» разбросу внутри классов. Тогда ЛДФ Фишера определяется как такой вектор $\mathbf{W}$, для которого критерий максимален:

$$J(\mathbf{W}) = \frac{(m_1 - m_2)^2}{s_1^2 + s_2^2}.$$

Для $\mathbf{W}$, найденного по данному критерию, классы, спроецированные на направление $\mathbf{W}$, максимально удалены друг от друга.

С учетом выражений (5.13) и (5.14) выше приведенный критерий записывается как

$$J(\mathbf{W}) = \frac{\mathbf{W}^T \mathbf{S}_B \mathbf{W}}{\mathbf{W}^T \mathbf{S}_W \mathbf{W}}.$$

Анализ этой формулы [89] показывает, что максимум $J(\mathbf{W})$ достигается при

$$\mathbf{W} = \mathbf{S}_W^{-1}(\mathbf{M}_1 - \mathbf{M}_2), \quad (5.15)$$

что почти совпадает с выражением (5.11) для нормальных классов с равными ковариационными матрицами.

Нахождение $\mathbf{W}$ сводит задачу описания классов в многомерном пространстве к их одномерному описанию, поэтому нахождение $\mathbf{W}$ можно рассматривать как задачу сокращения размерности. Однако, в этом случае проектирование классов на $\mathbf{W}$ и их статистический одномерный анализ позволяет найти наилучший порог a и сформулировать решающее правило.

Практическое задание к п. 5.3

Проведите анализ выборки данных, импортированных при выполнении практического задания 2 к главе 3, с использованием линейного дискриминантного анализа:

- 1) совместно с преподавателем выберите данные для анализа;
- 2) вычислите параметры биомедицинских сигналов, используя рекомендации, описанные в главе 4. В качестве альтернативных параметров могут использоваться главные компоненты, описанные в п. 5.2;

- 3) реализуйте алгоритм расчета линейных дискриминантных функций Фишера. Сравните полученное решение с функциями библиотеки `scikit-learn`;
- 4) оцените результат линейного дискриминантного анализа при варьировании используемых параметров биосигналов. Сравните эти результаты с результатами, полученными при использовании главных компонент в качестве параметров;
- 5) визуализируйте и сохраните полученный результат.

5.4. Машинное обучение нейрона

Одним из интересных примеров последовательного машинного обучения линейного классификатора является процедура машинного обучения одного формального нейрона — алгоритм персептрона, который моделирует биологический нейрон и является основой построения нейронных сетей. Это простейший алгоритм, работающий только на непересекающихся классах.

Рассмотрим задачу нахождения линейной разделяющей границы для двух классов объектов ω_1 и ω_2 . Предположим, что для обучения имеется N объектов обоих классов. Обозначим это множество объектов через Ψ . Множество Ψ состоит из двух обучающих выборок Ψ_1 (объекты первого класса ω_1) и Ψ_2 (объекты второго класса ω_2). Будем предполагать, что существует линейная функция, разделяющая Ψ_1 и Ψ_2 (т. е. Ψ_1 и Ψ_2 линейно разделимы).

Линейное решающее правило, разделяющее в n -мерном пространстве Ψ_1 и Ψ_2 , как было показано ранее, имеет вид (5.2)

$$\sum_{k=1}^n w_k x_k + w_{n+1} = 0,$$

где $x_1, x_2, \dots, x_n$ — компоненты вектора объекта $\mathbf{X} = (x_1, x_2, \dots, x_n)$. Добавим к ранее введенному вектору-объекту $\mathbf{X}$ $(n + 1)$ -ю компоненту, равную 1, и обозначим такой *расширенный* вектор-объект через $\mathbf{Y} = (y_1, y_2, \dots, y_{n+1})$, где $y_i = x_i$ для $i = 1, \dots, n$, $y_{n+1} = 1$. Тогда обучающим выборкам Ψ_1 и Ψ_2 будут соответствовать обучающие выборки ϑ_1 и ϑ_2 множества соответствующих расширенных векторов. Их объединение обозначим через ϑ .

В этом случае решающее правило будет иметь вид

$$\sum_{k=1}^{n+1} w_k y_k = 0$$

или в векторной форме $\mathbf{W}^T \mathbf{Y} = 0$.

Предположение о линейной разделимости Ψ_1 и Ψ_2 означает, что существует вектор $\mathbf{W}$, называемый решающим весовым вектором, такой, что $\mathbf{W}^T \mathbf{Y} > 0$ для каждого $\mathbf{Y} \in \vartheta_1$, $\mathbf{W}^T \mathbf{Y} < 0$ для каждого $\mathbf{Y} \in \vartheta_2$.

Если все образы из ϑ_2 умножить на -1 , то вышеприведенное условие запишется в виде $\mathbf{W}^T \mathbf{Y} > 0$ для каждого $\mathbf{Y} \in \vartheta_2$.

При таком представлении классификатор на два класса можно изобразить в виде рис. 5.14. Это — схема персептрона. Обучение такого классификатора на обучающем множестве ϑ заключается в подборе таких значений компонент весового вектора $\mathbf{W} = (w_1, w_2, \dots, w_{n+1})$, чтобы при предъявлении на входе любого $\mathbf{Y} \in \vartheta_1$ на выходе появлялось бы $+1$, а при появлении любого $\mathbf{Y} \in \vartheta_2$ на выходе было бы -1 .

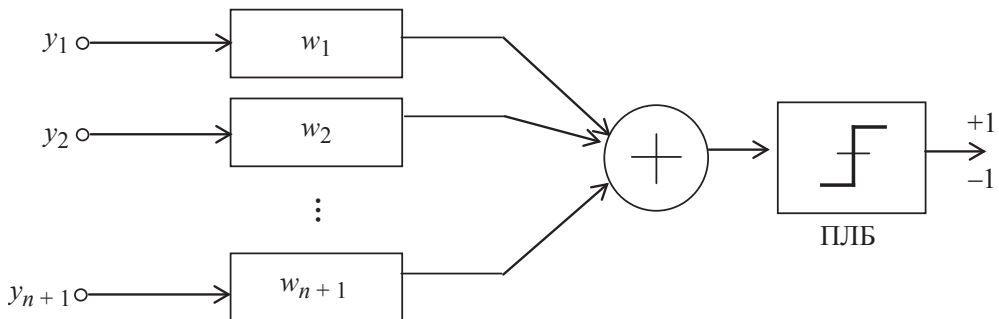


Рис. 5.14. Классификатор на 2 класса (персептрон)

Каждому $\mathbf{Y} \in \vartheta_2$ можно поставить в соответствие плоскость (плоскость объекта) в этом же пространстве, которая проходит через начало координат и перпендикулярна $\mathbf{Y}$. Ответ ПЛБ зависит от того, с какой стороны от плоскости объекта находится точка $\mathbf{W}$, как показано на рис. 5.15.

Будем считать, что при правильной классификации объектов из обучающей выборки ϑ (с учетом изменения знака на противоположный у всех объектов из ϑ_2) ответ ПЛБ должен быть всегда положителен. Тогда для N объектов $\mathbf{Y} \in \vartheta$ можно построить N плоскостей и по ним определить некоторую область решений, как показано на рис. 5.16.

Для любого $\mathbf{W}$ из этой области решений для всех объектов $\mathbf{Y} \in \mathcal{D}$ будет выполняться $\mathbf{W}^T \mathbf{Y} > 0$.

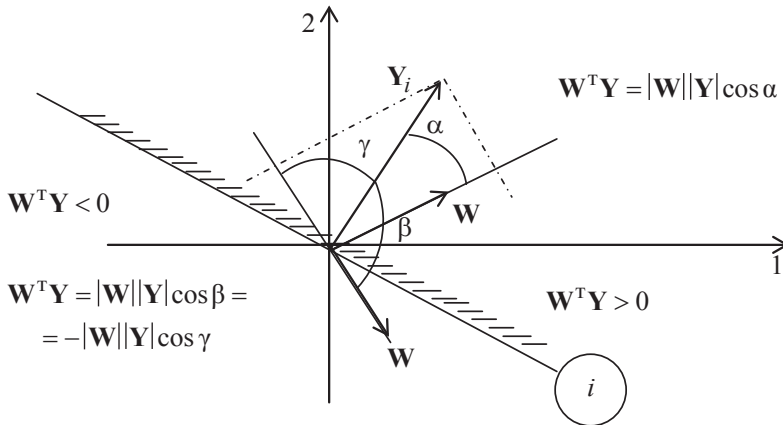


Рис. 5.15. Определение плоскости объекта для i -го объекта

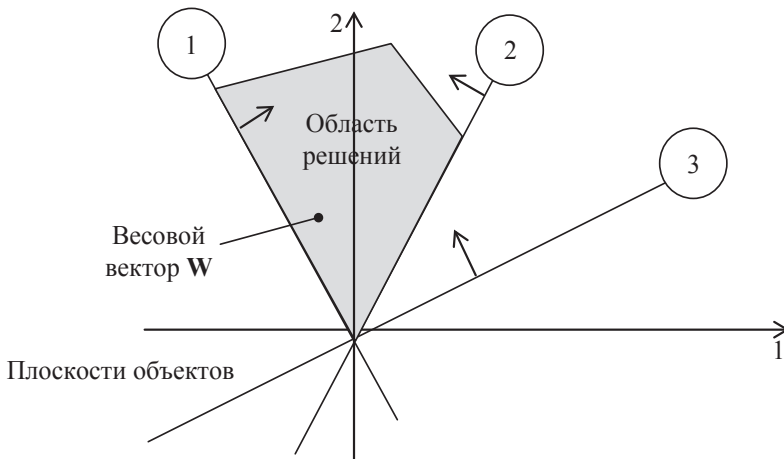


Рис. 5.16. Двумерное пространство весов с тремя плоскостями объектов

Процедуры последовательного обучения с коррекцией ошибок

Можно по N обучающим объектам найти область решений и выбрать $\mathbf{W}$ внутри этой области. Мы рассмотрим иную процедуру последовательного обучения [67; 91; 97], когда обучающие объекты предъявляются последовательно, по одному, а весовой вектор $\mathbf{W}$ уточняется с каждым новым входным вектором $\mathbf{Y}$. В этом случае, если после предъ-

явления объекта $\mathbf{Y} \in$ выполняется $\mathbf{W}^T \mathbf{Y} > 0$, то $\mathbf{W}$ остается неизменным. Если же $\mathbf{W}^T \mathbf{Y} < 0$, то вычисляется новое значение $\mathbf{W}$, равное $\mathbf{W}'$ по формуле

$$\mathbf{W}' = \mathbf{W} + c\mathbf{Y},$$

где $c > 0$ называется коэффициентом коррекции. Из этого выражения видно, что новое значение весового вектора выбирается на нормали к плоскости объекта в ту сторону, где $\mathbf{W}^T \mathbf{Y} > 0$, как показано на рис. 5.17, а и б.

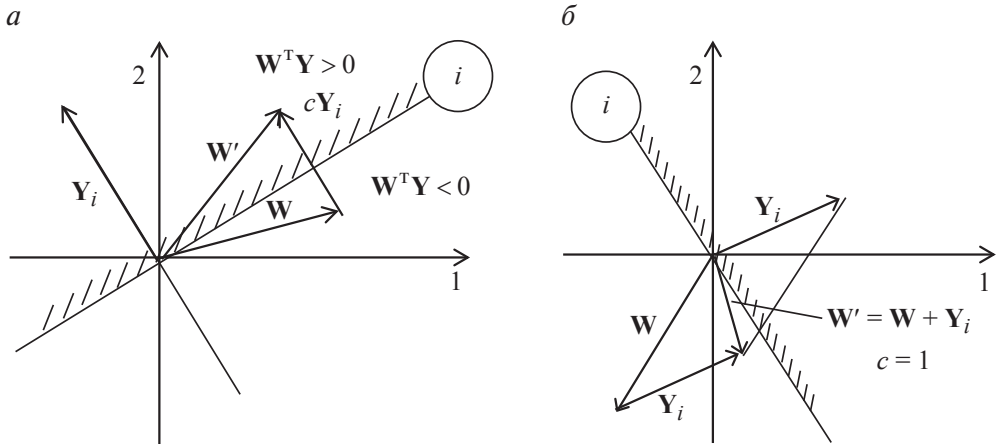


Рис. 5.17. Визуализация алгоритмов обучения:
а — последовательное обучение с коррекцией ошибок;
б — алгоритм с постоянным коэффициентом

Объекты предъявляются либо циклически, либо в случайном порядке до тех пор, пока не повторится результат испытания каждого из них. Начальное значение $\mathbf{W}$ произвольно.

Рассмотрим три варианта алгоритма обучения в зависимости от процедуры выбора величины c .

Если выбирается $c = 1$, то $\mathbf{W}' = \mathbf{W} + \mathbf{Y}$. Этот алгоритм с постоянным коэффициентом в зависимости от отношения величин $\mathbf{W}^T \mathbf{Y}$ и c исправляет или не исправляет ошибку, произошедшую на данном объекте, как показано на рис. 5.17, б, так как $\mathbf{W}$ перемещается всегда на фиксированную величину, которая может быть больше или меньше расстояния $\mathbf{W}$ до плоскости объекта.

Для алгоритма с абсолютной коррекцией величина c выбирается заново на каждом шаге предъявления объекта $\mathbf{Y}$, причем значение c вы-

бирается как наименьшее целое число, которое делает величину $\mathbf{W}^T \mathbf{Y}$ большей нуля, т. е. выбирается из условия

$$\mathbf{W}^T \mathbf{Y} = (\mathbf{W} + c\mathbf{Y})^T \mathbf{Y} > 0.$$

При этом $\mathbf{W}$ перемещается перпендикулярно плоскости объекта на величину, чуть большую расстояния от $\mathbf{W}$ до этой плоскости. Если $\mathbf{Y}$ расположен на плоскости объекта, то из вышеприведенного выражения следует, что

$$(\mathbf{W} + c\mathbf{Y})^T \mathbf{Y} = \mathbf{W}^T \mathbf{Y} + c\mathbf{Y}^T \mathbf{Y} = 0, \quad \mathbf{W}^T \mathbf{Y} = -c\mathbf{Y}^T \mathbf{Y}$$

$$|\mathbf{W}^T \mathbf{Y}| = |c\mathbf{Y}^T \mathbf{Y}| = c|\mathbf{Y}|^2, \quad c = \frac{|\mathbf{W}^T \mathbf{Y}|}{|\mathbf{Y}|^2}.$$

Поэтому c должно быть наименьшим целым числом, при этом большим, чем

$$\frac{|\mathbf{W}^T \mathbf{Y}|}{\mathbf{Y}^T \mathbf{Y}} = \frac{|\mathbf{W}^T \mathbf{Y}|}{|\mathbf{Y}|^2}.$$

После корректировки весов по этому правилу на данном шаге классификатор будет правильно распознавать предъявленный объект $\mathbf{Y}$ (рис. 5.18), где c — наименьшее целое, большее, чем $\frac{|\mathbf{W}^T \mathbf{Y}_i|}{|\mathbf{Y}_i^T \mathbf{Y}_i|}$; $|c\mathbf{Y}_i| = \frac{|\mathbf{W}^T \mathbf{Y}_i|}{|\mathbf{Y}_i|^2}$.

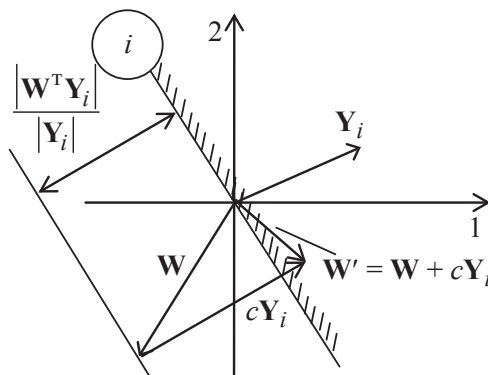


Рис. 5.18. Алгоритм с абсолютной коррекцией

Очевидно, что этот алгоритм приводит к тем же результатам, что и предыдущий алгоритм с постоянным коэффициентом при $c = 1$, если

при использовании этого алгоритма объект предъявляется повторно до тех пор, пока он не будет правильно распознан.

Алгоритм с частичной коррекцией перемещает весовой вектор на расстояние, пропорциональное расстоянию от точки до плоскости, т. е. величине $\frac{|\mathbf{W}^T \mathbf{Y}|}{|\mathbf{Y}|}$. Если новую точку выбирать на нормали к плоскости на расстоянии, пропорциональном этой величине, то

$$|\mathbf{W}' - \mathbf{W}| = \lambda \frac{|\mathbf{W}^T \mathbf{Y}|}{|\mathbf{Y}|},$$

где $\lambda > 0$ — коэффициент пропорциональности. Тогда

$$|\mathbf{Y}| |\mathbf{W}' - \mathbf{W}| = \lambda |\mathbf{W}^T \mathbf{Y}|.$$

Так как векторы $\mathbf{Y}$ и $\mathbf{W}' - \mathbf{W}$ коллинеарны, то из последнего равенства следует

$$|\mathbf{Y}^T (\mathbf{W}' - \mathbf{W})| = \lambda |\mathbf{W}^T \mathbf{Y}|.$$

Так как $\mathbf{W}' - \mathbf{W} = c\mathbf{Y}$ то $|\mathbf{W}' - \mathbf{W}| = |c\mathbf{Y}|$ и поэтому $|\mathbf{Y}| |c\mathbf{Y}| = \lambda |\mathbf{W}^T \mathbf{Y}|$ и $|c\mathbf{Y}| = \frac{\lambda |\mathbf{W}^T \mathbf{Y}|}{|\mathbf{Y}|}$ откуда $c = \frac{\lambda |\mathbf{W}^T \mathbf{Y}|}{|\mathbf{Y}|^2} = \frac{\lambda |\mathbf{W}^T \mathbf{Y}|}{\mathbf{Y}^T \mathbf{Y}}$.

Коэффициент λ выбирается в диапазоне $0 < \lambda < 2$. В случае $\lambda = 0$ весовой вектор вообще не изменяет положения, для $\lambda = 1$ вектор $\mathbf{W}$ переводится в $\mathbf{W}'$, расположенный непосредственно на плоскости объекта, а при $\lambda = 2$ весовая точка переводится по другую сторону плоскости на расстояние, равное исходному расстоянию $\mathbf{W}$ до плоскости (рис. 5.19, а), при условии, что $d = \frac{|\mathbf{W}^T \mathbf{Y}_i|}{|\mathbf{Y}_i|}$; $c = \lambda \frac{|\mathbf{W}^T \mathbf{Y}_i|}{|\mathbf{Y}_i|^2}$; $\lambda = 2$.

В общем случае алгоритм обучения с коррекцией ошибок заключается в последовательном испытании множества плоскостей объектов. Для каждой плоскости выясняется, лежит ли весовая точка по нужную сторону. Если это расположение неправильно, то весовая точка перемещается перпендикулярно плоскости по направлению к ней. Процесс продолжается до тех пор, пока не будет получен разделяющий весовой вектор (рис. 5.19, б). В теории доказывается [91], что все три алгоритма сходятся за конечное число шагов.

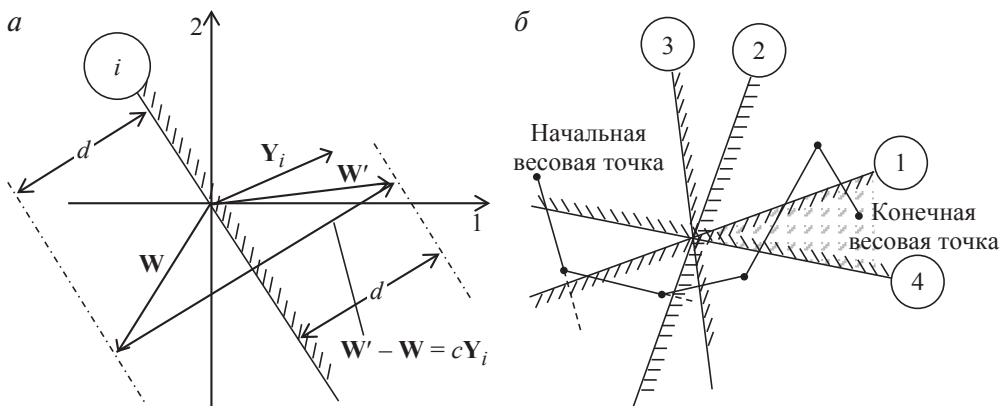


Рис. 5.19. Визуализация алгоритма обучения:

a — алгоритм с частичной коррекцией; *б* — процесс обучения с исправлением ошибок

Практическое задание к п. 5.4

Выполните анализ выборки данных, импортированных при выполнении практического задания 2 к главе 3, с использованием алгоритма персептрона:

- 1) совместно с преподавателем выберите выборку для анализа;
- 2) вычислите параметры биомедицинских сигналов, используя рекомендации, описанные в главе 4. В качестве альтернативных параметров могут использоваться главные компоненты, описанные в п. 5.2;
- 3) реализуйте алгоритм обучения простейшей нейронной сети. Сравните полученное решение с функциями библиотеки `scikit-learn`;
- 4) визуализируйте и сохраните полученный результат.

Контрольные вопросы к главе 5

1. Опишите методику использования разделяющей поверхности в признаковом пространстве.
2. Опишите методику использования классификатора по минимуму расстояния.

3. Охарактеризуйте метод главных компонент. Опишите принцип уменьшения размерности пространства признаков.
4. Опишите методику применения линейного дискриминантного анализа.
5. Охарактеризуйте обучение простейшей нейронной сети. Сравните процедуры обучения с коррекцией ошибок.

Глава 6.

Обработка изображений в биомедицинских задачах*

Мощность современных процессоров персональных компьютеров, видеокамер, графических средств визуализации и, главное, развитый математический аппарат и наличие специализированных библиотек подпрограмм способны обеспечить практически любые запросы по обработке и анализу медицинских изображений. Развитие цифровых технологий за последнее время привело к появлению большого количества новых методов диагностики и визуализации. У врача-диагноста появились новые возможности целенаправленно воздействовать на процесс визуализации медицинского изображения для качественной и своевременной диагностики.

Эта глава посвящена математическим и алгоритмическим основам современных методов цифровой обработки изображений. Здесь также освещены современные и популярные сегодня алгоритмы решения основных задач улучшения и интеллектуального анализа изображений.

Изображения, поступающие из цифровой видеокамеры или фотоаппарата, представляют собой двумерный массив $f(x, y)$ — матрицу из M строк и N столбцов, где (x, y) — дискретные координаты. Для ясности обозначений и большего удобства используются целочисленные значения этих координат: $x = 0, 1, 2, \dots, M-1$ и $y = 0, 1, 2, \dots, N-1$, принимая за начало координат левый верхний угол изображения, где $(x, y) = (0, 0)$. Следующим значением координат вдоль первой строки изображения будет точка $(x, y) = (0, 1)$. В общем случае значение изображения в произвольной координатной точке (x, y) обозначается $f(x, y)$, где x и y — целые числа. Область действительной координат-

* Автор главы 6: В. Б. Костоусов (Институт математики и механики имени Н. Н. Красовского Уральского отделения Российской академии наук, г. Екатеринбург)

ной плоскости, охватываемая координатами изображения, называется *пространственной областью*, а x и y — *пространственными переменными* или *пространственными координатами*.

Представление изображения в виде числовой матрицы размерами $M \times N$ можно записать в форме соотношения

$$f(x, y) = \begin{bmatrix} f(0, 0) & f(0, 1) & \dots & f(0, N-1) \\ f(1, 0) & f(1, 1) & \dots & f(1, N-1) \\ \vdots & \vdots & \dots & \vdots \\ f(M-1, 0) & f(M-1, 1) & \dots & f(M-1, N-1) \end{bmatrix}.$$

Здесь в правой части стоит матрица действительных чисел. Каждый элемент этой матрицы называется *элементом изображения*, или *пикселем*. Далее повсюду будут употребляться термины *изображение* и *пиксель* для указания на цифровое изображение и его элементы. Следует отметить, что начало координат расположено в левом верхнем углу цифрового изображения, ось x направлена вниз, а ось y — вправо.

Если значения яркости в результате *квантования* становятся целочисленными, тогда вместо множества *действительных чисел* используется *целочисленная шкала*, начиная от 0 и до $L-1$. Для вычислений или разработки алгоритмов часто бывает полезно отобразить шкалу L значений яркости на интервал $[0, 1]$, так что они перестают быть целочисленными. Однако при хранении и визуализации изображений в большинстве случаев полученные значения масштабируются обратно на интервал целых значений $[0, L-1]$. Для значений M и N далее не устанавливаются специальных ограничений помимо того, что они должны быть положительными целочисленными значениями. Однако значение L по соображениям удобства построения оборудования для хранения и квантования обычно оказывается равным целочисленной степени двойки: $L = 2^k$.

Предполагается, что дискретные уровни яркости расположены с постоянным шагом (т. е. используется равномерное квантование) и принимают целые значения в интервале $[0, L-1]$. Иногда интервал значений яркости неформально называют *динамическим диапазоном* изображения. Здесь определяется *динамический диапазон* системы регистрации изображений как отношение максимально возможного значения измеренной яркости к минимальному уровню яркости, обнаруживаемому этой системой. Как правило, верхний предел опре-

деляется *насыщением*, а нижний предел — *шумом*. По существу, динамический диапазон устанавливает минимальный и максимальный уровни яркости, которые система способна представлять и, следовательно, могут присутствовать в изображении. С этим понятием тесно связан *контраст* изображения, который определяется как разность между максимальным и минимальным уровнями яркости в данном изображении. Если заметная доля пикселей обладает большим динамическим диапазоном, можно ожидать, что такое изображение имеет высокий контраст. Наоборот, изображение с малым динамическим диапазоном обычно выглядит тусклым, размытым и серым.

В этой главе рассмотрены следующие вопросы:

- базовые операции обработки изображений, включая такие операции в пространственной области, как свертка и корреляция, эквализация гистограммы, пороговая обработка и детектирование контуров;
- пространственные и частотные методы фильтрации и повышение резкости изображений;
- методы восстановления и реконструкции изображений, включая фильтр Винера и метод регуляризации Тихонова для решения обратных некорректных задач;
- продвинутые дескрипторы особенностей, применяемые в современных системах компьютерного зрения, включая системы анализа изображений в медицине;
- современные методы обработки видеоизображений и их применение в задаче мониторинга пациента в палате интенсивной терапии;
- методы решения задачи дистанционной фотоплетизмографии на примере реальных данных мониторинга пациента в палате интенсивной терапии.

6.1. Базовые операции

Пространственные операции

Пространственные операции отличаются тем, что они выполняются непосредственно над значениями пикселей данного изображения. Эти операции можно разделить на три широких класса: поэлемент-

ные операции (поточечная обработка), операции над окрестностью (локальная обработка) и геометрические преобразования (глобальная обработка).

Поэлементные операции

Простейшая операция над цифровым изображением состоит в изменении значений отдельных пикселей исходя из их яркости. Такой способ обработки может быть выражен функцией градационного преобразования (также называемой функцией преобразования интенсивностей или функцией отображения) вида

$$s = T(z),$$

где z — яркость пикселя исходного изображения, а s — полученная после преобразования яркость соответственного пикселя на обработанном изображении.

Операции над окрестностью

Обозначим S_{xy} множество точек окрестности произвольной точки (x, y) изображения f . При обработке окрестности генерируется соответственный (имеющий те же координаты) пиксель в выходном изображении g таким образом, что его значение определяется с помощью заданной операции над пикселями исходного изображения с координатами из S_{xy} . Например, пусть эта операция состоит в вычислении среднего значения яркости пикселей прямоугольной окрестности с размерами $m \times n$ и центром в точке (x, y) . Координаты пикселей в этой области составляют множество S_{xy} . Рис. 6.1 иллюстрирует такой процесс.

Эту операцию можно выразить уравнением

$$g(x, y) = \frac{1}{mn} \sum_{(r, c) \in S_{xy}} f(r, c),$$

где r и c есть координаты строки и столбца для тех пикселей, координаты которых входят в множество S_{xy} . Изображение g формируется путем варьирования координат (x, y) так, что центр окрестности передвигается по изображению f от пикселя к пикселю и в каждом новом месте повторяется та же самая операция над окрестностью.

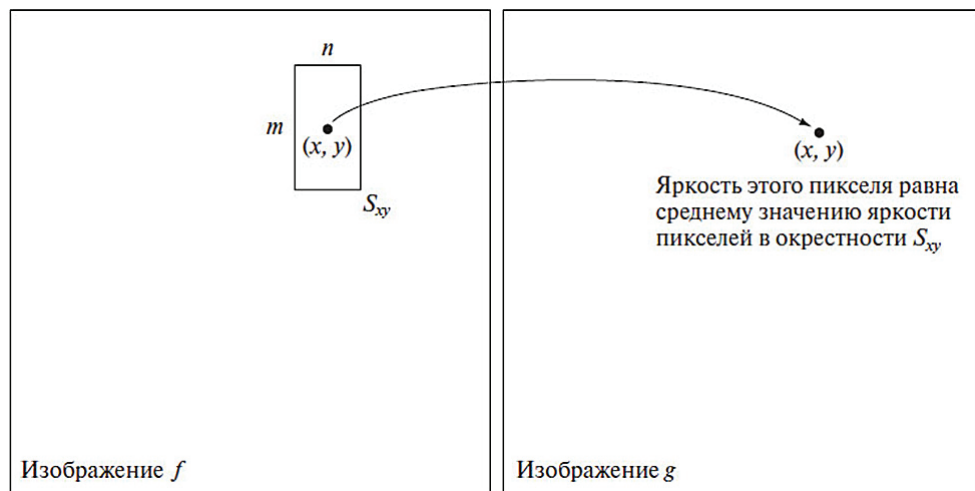


Рис. 6.1. Локальное усреднение с помощью обработки окрестности

В случае маски размерами $m \times n$ будем полагать, что $m = 2a + 1$ и $n = 2b + 1$, где a и b есть неотрицательные целые. Это означает, что в дальнейшем будут рассматриваться маски нечетных размеров, причем наименьшей будет маска 3×3 элемента. В общем виде фильтрация изображения размерами $M \times N$ с помощью фильтра размерами $m \times n$ задается выражением

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

где x и y изменяются так, что каждая точка оператора w в какой-то момент попадает на каждый пиксель изображения f .

Геометрические преобразования

Геометрические преобразования изменяют пространственные взаимосвязи между пикселями на изображении. Геометрические преобразования часто называют преобразованиями *резинового листа*, поскольку их можно представить себе как процесс распечатки изображения на листе из резины и дальнейшее растягивание этого листа в соответствии с определенными правилами. С точки зрения цифровой обработки изображений геометрические преобразования состоят из следующих двух основных операций: пространственного преобразования

координат и интерполяции значений яркости, при которой происходит присвоение значений яркости точкам изображения, подвергнутого пространственному преобразованию.

Пространственная корреляция и свертка

Существуют две близкородственные концепции, которые при выполнении линейной пространственной фильтрации необходимо четко понимать. Одна из них — *корреляция*, а другая — *свертка*. Корреляция есть процесс движения маски фильтра по изображению и вычисление суммы произведений значений элементов маски и значений пикселей, на которые попадают соответствующие элементы маски, для всех точек изображения. Механизмы свертки такие же, но за исключением того, что предварительно маска фильтра поворачивается на 180° .

Корреляция фильтра $w(x, y)$ размерами $m \times n$ с изображением $f(x, y)$, обозначаемая как $w(x, y) \bullet f(x, y)$, задается уравнением, записанным выше, которое повторим здесь для удобства:

$$w(x, y) \bullet f(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x + s, y + t). \quad (6.1)$$

Это уравнение вычисляется для всех значений переменных смещения x и y , так что все элементы из w попадают на каждый из пикселей f , при этом предполагается, что f расширено соответствующим образом. Как объяснено ранее, $a = (m - 1)/2$, $b = (n - 1)/2$ и для удобства обозначений предполагается, что m и n — целые нечетные.

Подобным же образом свертка $w(x, y)$ с $f(x, y)$, обозначаемая как $w(x, y) * f(x, y)$, задается уравнением

$$w(x, y) * f(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x - s, y - t), \quad (6.2)$$

где знак минус действует как смена порядка на обратный (т. е. поворот на 180°). Переворот и сдвиг f вместо w сделан для упрощения обозначений, а также следуя общепринятым правилам. Результат тот же самый. Как и в случае корреляции, это уравнение вычисляется для всех значений переменных смещения x и y , так что все элементы из w попадают на каждый из пикселей f , расширенного соответствующим образом.

Эквализация гистограммы

Гистограммой цифрового изображения с уровнями яркости в диапазоне $[0, L - 1]$ называется дискретная функция $h(r_k) = n_k$, где r_k есть k -й уровень яркости, а n_k — число пикселей на изображении, имеющих яркость r_k . Общей практикой является нормировка гистограммы путем деления каждого из ее значений на общее число пикселей в изображении, обозначаемое произведением MN , где, как обычно, M и N есть число строк и столбцов изображения. Тем самым значения нормированной гистограммы будут равны $p(r_k) = n_k/MN$ для $k = 0, 1, \dots, L - 1$. Можно сказать, что $p(r_k)$ есть оценка вероятности появления пикселя со значением яркости r_k . Следует заметить, что сумма всех значений нормированной гистограммы равна единице.

Гистограммы являются основой для многочисленных методов пространственной обработки изображений.

Для простоты сначала рассматриваются непрерывные значения яркостей, и пусть переменная r означает яркости элементов обрабатываемого изображения. Как обычно, при этом предполагается, что r распределена в диапазоне $[0, L - 1]$, причем значение $r = 0$ соответствует черному, а $r = L - 1$ — белому. Для любого r , удовлетворяющего вышеуказанным условиям, рассматривается преобразование (отображение яркости) вида

$$s = T(r) \quad 0 \leq r \leq L - 1,$$

которое для любого пикселя, имеющего значение r , дает значение s . Подразумевается, что функция преобразования $T(r)$ удовлетворяет следующим условиям:

- а) $T(r)$ является монотонно неубывающей функцией на интервале $0 \leq r \leq L - 1$; а также
- б) $0 \leq T(r) \leq L - 1$ при $0 \leq r \leq L - 1$.

В некоторых формулировках, рассмотренных ниже, используется обратное преобразование

$$r = T^{-1}(s) \quad 0 \leq s \leq L - 1,$$

в случае которого условие а) изменяется на а') $T(r)$ является монотонно возрастающей функцией на интервале $0 \leq r \leq L - 1$.

В условии а) требование монотонного возрастания функции $T(r)$ гарантирует, что возрастание яркости на входе не может привести

к убыванию яркости на выходе, предотвращая тем самым артефакты, вызываемые инвертированием изменения яркости. Условие б) означает, что допустимый диапазон выходных значений сигнала совпадает с диапазоном входных значений. Наконец условие а') гарантирует, что обратное отображение из s в r будет взаимно однозначным, предотвращая неопределенности.

Уровни яркости на изображении могут рассматриваться как значения случайной величины в интервале $[0, L-1]$. Важнейшей характеристикой случайной величины является плотность распределения вероятностей (ПРВ). Пусть $p_r(r)$ и $p_s(s)$ означают ПРВ случайных переменных r и s соответственно, где индекс при p означает, что $p_r(r)$ и $p_s(s)$ являются, вообще говоря, разными функциями. Из элементарной теории вероятностей следует, что если $p_r(r)$ и $T(r)$ известны и $T(r)$ является непрерывной и дифференцируемой на множестве интересующих значений, то ПРВ результата преобразования (отображения) — переменной s — может быть получена с помощью следующей простой формулы:

$$p_s(s) = p_r(r) \left| \frac{dr}{ds} \right|. \quad (6.3)$$

Таким образом, ПРВ значений преобразованного сигнала s задается через ПРВ значений яркостей входного изображения и выбранную функцию преобразования (напомним, что r и s связаны функцией $T(r)$).

В обработке изображений особую важность имеет следующая функция:

$$s = T(r) = (L-1) \int_0^r p_r(w) dw, \quad (6.4)$$

где w — переменная интегрирования. Правая часть данного уравнения есть не что иное, как функция распределения (ФР) случайной переменной r . Поскольку ФР всегда положительна, а интеграл функции равен площади под графиком функции, следовательно, функция преобразования в уравнении (6.4) удовлетворяет условию а), т. к. площадь под графиком функции не может уменьшаться при увеличении r . Когда достигается верхний предел значений $r = L - 1$, интеграл становится равным 1 (площадь под кривой ПРВ всегда равна 1), а максимальное значение s становится равным $(L - 1)$; значит условие б) также выполняется.

Зная функцию преобразования $T(r)$, ПРВ $p_s(s)$ можно найти из уравнения (6.3). Из дифференциального исчисления известно, что произ-

водная определенного интеграла по его верхнему пределу равна подынтегральному выражению в точке верхнего предела (правило Лейбница). Другими словами,

$$\frac{ds}{dr} = \frac{dT(r)}{dr} = (L-1) \frac{d}{dr} \left[\int_0^r p_r(w) dw \right] = (L-1) p_r(r).$$

Подставляя этот результат для dr/ds в уравнение (6.3) и предполагая, что все значения плотности вероятностей больше нуля, получаем в результате

$$p_s(s) = p_r(r) \left| \frac{dr}{ds} \right| = p_r(r) \left| \frac{1}{(L-1) p_r(r)} \right| = \frac{1}{(L-1)} \quad 0 \leq s \leq L-1. \quad (6.5)$$

Таким образом, получилось, что $p_s(s)$ есть *равномерная* плотность распределения вероятностей. Попросту говоря, было продемонстрировано, что выполнение градиационного преобразования согласно функции, заданной уравнением (6.4), приводит к получению некоторой случайной величины s , характеризующейся равномерной ПРВ. Здесь важно заметить, что хотя $T(r)$, как это следует из (6.4), зависит от $p_r(r)$, результирующая ПРВ $p_s(s)$, как следует из (6.5), всегда является равномерной, независимо от формы $p_r(r)$.

В случае дискретных значений вместо плотностей распределения вероятностей и интегралов имеются значения гистограмм и сумм. Как упоминалось выше, вероятность появления на цифровом изображении пикселя со значением яркости r_k приближенно равна

$$p_r(r_k) = \frac{n_k}{MN} \quad k = 0, 1, 2, \dots, L-1,$$

где MN есть общее число пикселей на изображении, n_k — число точек яркости r_k , а L — максимально допустимое число уровней яркости на изображении (т. е. 256 для 8-битового изображения). Как отмечалось в начале главы, зависимость $p_r(r_k)$ обычно называют *гистограммой*.

Дискретным аналогом функции преобразования, задаваемой уравнением (6.4), будет

$$s_k = T(r_k) = (L-1) \sum_{j=0}^k p_r(r_j) = \frac{(L-1)}{MN} \sum_{j=0}^k n_j \quad k = 0, 1, 2, \dots, L-1. \quad (6.6)$$

Таким образом, обработанное (выходное) изображение получается отображением каждого пикселя входного изображения, имеющего яр-

кость r_k , в соответствующий элемент выходного изображения со значением s_k , согласно уравнению (6.6). Преобразование (отображение), задаваемое уравнением (6.6), называется *эквализацией*, или *линеаризацией* гистограммы.

Поскольку гистограмма есть приближение ПРВ и в процессе преобразования новые уровни яркости не образуются, а могут лишь сливаться, получение в результате эквализации совершенно ровной гистограммы является на практике весьма редким случаем. Таким образом, в отличие от непрерывного случая, в общем случае нельзя доказать, что дискретная эквализация гистограммы приводит к равномерной гистограмме. Однако использование уравнения (6.6) имеет тенденцию к растягиванию гистограммы исходного изображения и уровни яркости эквализованного изображения занимают более широкий диапазон шкалы яркостей. В итоге результатом является повышение контрастов.

Рассмотренный только что метод, кроме того, что повышает контраст изображения, обладает тем дополнительным преимуществом, что является полностью «автоматическим». Иными словами, получая на вход изображение, процедура эквализации гистограммы сводится к выполнению преобразования по формуле (6.6), опираясь лишь на информацию, которая может быть извлечена непосредственно из обрабатываемого изображения без указания каких-либо дополнительных параметров. Стоит также отметить простоту вычислений, которые требуются для реализации этого метода.

Пороговая обработка

Пороговые преобразования занимают центральное место в прикладных задачах сегментации изображений благодаря интуитивно понятным свойствам, простоте реализации и скорости вычислений.

Предположим, что показанная на рис. 6.2 гистограмма соответствует некоторому изображению $f(x, y)$, содержащему светлые объекты на темном фоне. В этом случае яркости пикселей объекта и фона сосредоточены вблизи двух преобладающих значений. Очевидный способ выделения объектов из окружающего фона состоит в выборе значения порога T , разграничивающего моды распределения яркостей. Тогда любая точка (x, y) , в которой $f(x, y) > T$, называется *точкой объекта*, а в противном случае — *точкой фона*. Другими слова-

ми, сегментированное изображение $g(x, y)$ задается соотношением (рис. 6.2)

$$g(x, y) = \begin{cases} 1, & \text{если } f(x, y) > T \\ 0, & \text{иначе.} \end{cases}$$

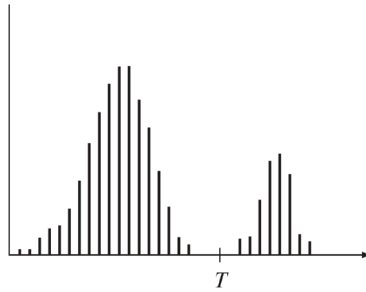


Рис. 6.2. Гистограмма яркости, допускающая разделение с помощью порога

Если значение T есть константа, применяемая для всего изображения, то эта формула описывает так называемое глобальное пороговое преобразование. Если величина порога не постоянна на изображении, говорят о преобразовании с переменным порогом. Говоря о пороговом преобразовании, иногда используют термины «локальное» или «в окрестности», чтобы указать, что величина переменного порога T в любой точке изображения (x, y) зависит от свойств окрестности (x, y) (например от средней яркости пикселей в этой окрестности). Если порог T зависит от характеристик обрабатываемого изображения и меняется с изменением пространственных координат (x, y) , то такое преобразование с переменным порогом часто называют динамическим или адаптивным.

Исходя из обсуждения выше, можно сделать интуитивный вывод, что успех порогового преобразования напрямую связан с шириной и глубиной впадин между модами распределения яркости. В свою очередь основные факторы, влияющие на характеристики впадин, это:

- 1) расстояние между пиками на гистограмме (чем оно больше, тем больше шансов разделить моды);
- 2) уровень шума в изображении (поскольку по мере увеличения шума моды распределения становятся шире);
- 3) соотношение размеров объектов и области фона;
- 4) равномерность освещения;
- 5) однородность яркости объектов и фона.

На практике для определения порога часто применяется *метод Оцу* [99], который является оптимальным в том смысле, что он минимизирует *межклассовую дисперсию* — хорошо известную характеристику, применяемую в статистическом дискриминантном анализе. Помимо оптимальности, метод Оцу имеет то важное свойство, что он целиком основан на вычислениях, выполняемых над гистограммой изображения, легко вычисляемым одномерным массивом.

Обнаружение разрывов яркости

В этом пункте описывается обнаружение резких *локальных* изменений яркости. *Пиксели перепада* — это те пиксели изображения, в которых функция яркости резко изменяется, а *перепад* (или *участок перепада*) — это связное множество пикселей перепада. *Детекторы перепадов* представляют собой методы локальной обработки изображения, направленные на обнаружение пикселей перепада.

При обнаружении контурных перепадов яркости выполняются три фундаментальных шага.

1. Сглаживание изображения с целью уменьшения шума.
2. *Обнаружение точек перепада*. Как отмечалось выше, это локальная операция, выделяющая на изображении все точки, которые могут быть точками перепада.
3. *Локализация перепада*. Цель этого шага состоит в том, чтобы выбрать среди выделенных кандидатов лишь те точки, которые действительно входят во множество точек, вместе составляющих контурный перепад.

Для нахождения в точке (x, y) изображения f одновременно величины перепада яркости и его направления обычно применяют градиент изображения ∇f , определяемый как *вектор*

$$\nabla f \equiv \text{grad}(\mathbf{f}) \equiv \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}.$$

Вектор градиента имеет важное геометрическое свойство: его направление совпадает с направлением максимальной скорости изменения функции f в точке (x, y) .

Модуль (длина) вектора ∇f обозначается $|\nabla f|$ и равен

$$|\nabla f| = \sqrt{g_x^2 + g_y^2} \quad (6.7)$$

Заметим, что g_x , g_y и $|\nabla f|$ представляют собой изображения тех же размеров, что и исходное изображение f ; они формируются, когда координаты x и y пробегают все пиксели f . Здесь операции сложения, возведения в квадрат и извлечения квадратного корня являются *поэлементными операциями*.

Направление вектора градиента задается углом

$$\alpha(x, y) = \arctg\left(\frac{g_x}{g_y}\right) \quad (6.8)$$

между направлением вектора ∇f в точке (x, y) и осью x . Как и изображение градиента, $\alpha(x, y)$ также является изображением тех же размеров, что и исходное f , и формируется (поэлементной) операцией арктангенса от (поэлементного) деления изображения g_y на изображение g_x . Направление контура в произвольной точке (x, y) *перпендикулярно* направлению $\alpha(x, y)$ вектора градиента в этой точке.

Простые методы обнаружения контуров

Вычисление градиента изображения состоит в получении величин частных производных $\partial f / \partial x$ и $\partial f / \partial y$ для каждой точки. В дискретном случае требуется найти дискретную аппроксимацию частных производных в окрестности данной точки:

$$g_x = \frac{\partial f(x, y)}{\partial x} = f(x+1, y) - f(x, y),$$

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

Вычисления по этим двум уравнениям для всех нужных значений x и y можно реализовать путем фильтрации изображения $f(x, y)$ с помощью простых одномерных масок.

Для обнаружения перепадов, идущих в диагональных направлениях, требуется применять двумерные маски. *Перекрестный градиентный оператор Робертса* [100] является одной из первых попыток примене-

ния двумерных масок с предпочитаемым диагональным направлением. Рассмотрим показанную на рис. 6.3, *a* окрестность 3×3 некоторого элемента изображения.

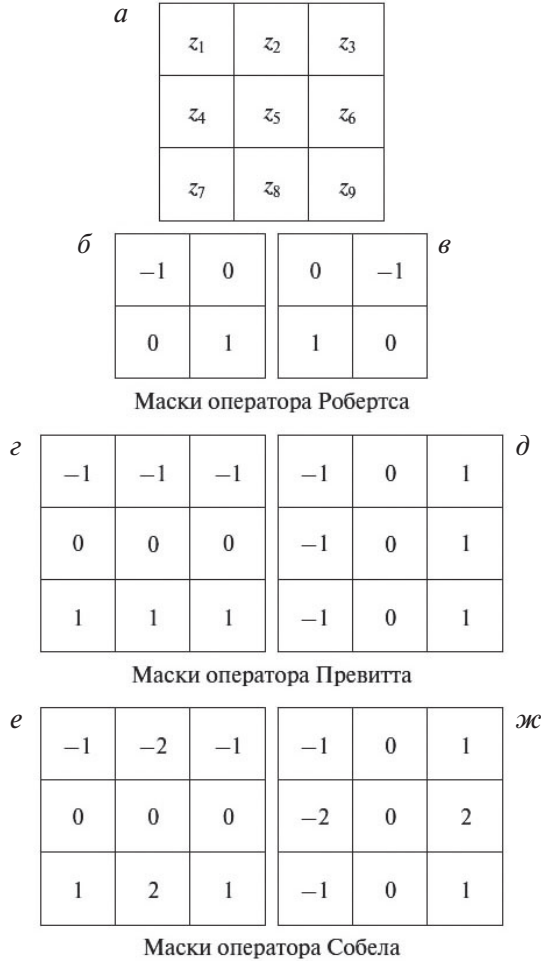


Рис. 6.3. Окрестность 3×3 внутри изображения (переменные z_i есть значения яркости) и различные маски, применяемые для вычисления градиента в центральной точке окрестности

Оператор Робертса вычисляет разности по диагонали:

$$g_x = \frac{\partial f}{\partial x} = (z_9 - z_5),$$

$$g_y = \frac{\partial f}{\partial y} = (z_8 - z_6).$$

В оставшейся части главы неявно предполагается, что f является функцией двух переменных, опуская эти переменные для простоты обозначений.

Эти производные могут быть реализованы путем фильтрации всего изображения с помощью масок, представленных на рис. 6.3, б, в.

Маски размерами 2×2 концептуально просты, но не столь удобны для вычисления направлений перепадов, как маски, симметричные относительно центрального элемента, минимальные размеры которых 3×3 . Такие маски учитывают значения с обеих сторон от центральной точки и поэтому дают больше информации о направлении перепада яркости. Простейшее дискретное приближение частных производных в случае использования масок 3×3 задается выражениями

$$g_x = \frac{\partial f}{\partial x} = (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3),$$

$$g_y = \frac{\partial f}{\partial y} = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7).$$

В этих формулах разность между суммами по нижней и верхней строкам окрестности 3×3 является приближенным значением производной по оси x , а разность между суммами по правому и левому столбцам этой окрестности — производной по оси y . Для реализации выше приведенных формул используется оператор, описываемый масками на рис. 6.3, г и д, который называется *оператором Превитта* [101].

Небольшое видоизменение последних двух формул состоит в использовании весового коэффициента 2 для средних элементов:

$$g_x = \frac{\partial f}{\partial x} = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3),$$

$$g_y = \frac{\partial f}{\partial y} = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7).$$

Это увеличенное значение используется для уменьшения эффекта сглаживания за счет придания большего веса средним точкам. Для реализации двух последних выражений используются маски на рис. 6.3, е и ж, отвечающие *оператору Собела* [102].

Маски оператора Превитта проще реализовать, чем маски оператора Собела, однако у последнего оператора влияние шума угловых элементов несколько меньше, что существенно при работе с производными. Отметим, что у каждой из масок на рис. 6.3 сумма коэффи-

циентов равна нулю, т. е. эти операторы будут давать нулевой отклик на областях постоянной яркости, как и следовало ожидать от дифференциального оператора.

Рассмотренные выше маски применяются для получения составляющих градиента g_x и g_y в каждом пикселе изображения, после чего эти частные производные используются для оценки величины перепада и направления контура. Для вычисления модуля и направления градиента эти составляющие необходимо использовать совместно согласно формулам (6.7) и (6.8).

Маски фильтров, применяемые для вычисления частных производных при нахождении градиента, часто называют *градиентными операторами*, *разностными операторами*, *операторами выделения контуров* или *детекторами контуров*.

Более совершенные методы обнаружения контуров

Рассмотренные выше методы выделения контуров использовали просто фильтрацию изображения с помощью одной или более масок без каких-либо предположений касательно характеристик контура и содержания шума. Здесь будут обсуждаться более совершенные методы, в которых делается попытка улучшения простых методов обнаружения контуров благодаря учету таких факторов, как шум в изображении и особенности самих контуров.

Детектор контуров Марра — Хилдрета

Одна из первых успешных попыток внедрения более сложного анализа в процесс нахождения контуров принадлежит Д. Марру и Е. Хилдрету [103]. Эти авторы на основе анализа требований к детекторам контуров пришли к выводу, что используемый для обнаружения контуров оператор должен обладать двумя характерными свойствами. Первое и главное, это должен быть дифференциальный оператор, способный вычислять приближенное значение первой или второй производной в каждой точке изображения. Во-вторых, он должен допускать настройку на любой желаемый масштаб, чтобы операторы большого размера можно было использовать для обнаружения размытых перепадов яркости, а операторы малого размера — для мелких деталей

с высокой резкостью. Оказалось, что наилучшим оператором, удовлетворяющим этим условиям, является фильтр $\nabla^2 G$, где ∇^2 — оператор Лапласа ($\partial^2/\partial x^2 + \partial^2/\partial y^2$), а G — двумерная гауссова функция:

$$G(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (6.9)$$

со стандартным отклонением σ . Чтобы найти выражение для $\nabla^2 G$, выполним следующие дифференцирования:

$$\begin{aligned} \nabla^2 G(x, y) &= \frac{\partial^2 G(x, y)}{\partial x^2} + \frac{\partial^2 G(x, y)}{\partial y^2} = \frac{\partial}{\partial x} \left(\frac{-x}{\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \right) + \frac{\partial}{\partial y} \left(\frac{-y}{\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \right) = \\ &= \left(\frac{x^2}{\sigma^4} - \frac{1}{\sigma^2} \right) e^{-\frac{x^2+y^2}{2\sigma^2}} + \left(\frac{y^2}{\sigma^4} - \frac{1}{\sigma^2} \right) e^{-\frac{x^2+y^2}{2\sigma^2}}. \end{aligned}$$

Приводя подобные члены, окончательно получаем:

$$\nabla^2 G(x, y) = \left(\frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} \right) e^{-\frac{x^2+y^2}{2\sigma^2}}. \quad (6.10)$$

Эту формулу обычно называют *лапласианом гауссиана* (ЛГ).

На рис. 6.4, *a–в* показаны трехмерный график, изображение и профиль ЛГ-функции в перевернутом виде (т. е. с обратным знаком).

Заметим, что ЛГ-функция пересекает нулевой уровень в точках, удовлетворяющих уравнению $x^2 + y^2 = 2\sigma^2$, что определяет окружность радиуса $\sqrt{2}\sigma$ с центром в начале координат. Из-за характерной формы графика на рис. 6.4, *a* лапласиан гауссиана иногда называют оператором вида *мексиканская шляпа*. На рис. 6.4, *г* приведена маска 5×5 , используемая в качестве приближения функции на рис. 6.4, *a* (на практике используют маску, имеющую значения с обратным знаком). Такое приближение не является единственно возможным; цель его — передать сущность *формы* ЛГ-функции, т. е. положительного всплеска в центре, окруженного областью отрицательных значений, которые, достигнув минимума, в дальнейшем по мере удаления от центра возрастают, приближаясь за пределами окрестности к нулевому значению. Сумма коэффициентов должна равняться нулю, чтобы маска давала нулевой отклик на области постоянной яркости.

Маски ЛГ можно строить любых желаемых размеров, выбирая шаг дискретизации уравнения (6.10) и масштабируя затем все коэффициенты так, чтобы их сумма равнялась нулю. Более эффективен следу-

ющий метод построения ЛГ-фильтра. Осуществляется дискретизация гауссовой функции (6.9) на сетке желаемых размеров $n \times n$, после чего вычисляется свертка полученного массива с маской лапласиана, например:

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}. \quad (6.11)$$

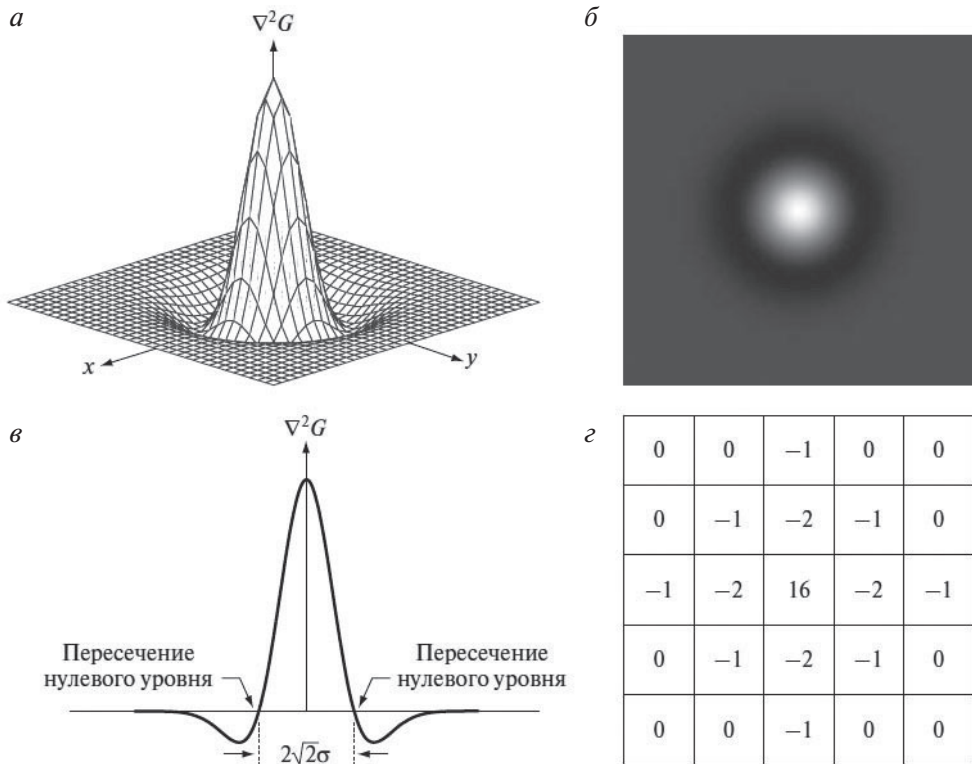


Рис. 6.4. Детектор контуров Марра — Хилдрета:

а — трехмерный график лапласиана гауссиана с *обратным знаком*; *б* — негативное изображение ЛГ-функции; *в* — поперечный профиль с отмеченными точками пересечения нулевого уровня; *г* — маска 5×5 для приближенного вычисления зависимости *а*

Поскольку свертка изображения с маской, у которой сумма коэффициентов равна нулю, дает результат также с нулевой суммой элементов, это автоматически обеспечивает выполнение того требования, что сумма коэффициентов ЛГ-фильтра должна быть равна нулю.

Выбор оператора $\nabla^2 G$ основывается на двух фундаментальных идеях. Первая — то, что гауссова функция в лапласиане гауссиана обеспечивает сглаживание изображения, тем самым снижая интенсивность структур (в том числе шумовых) с размерами меньше σ . Важным является то, что гауссова функция является гладкой не только в пространственной, но и в частотной областях (см. далее), что снижает вероятность появления артефактов (типа «звона»), которых не было в исходном изображении. Вторая идея связана с ∇^2 — второй производной в составе фильтра. Хотя для обнаружения скачков яркости могут применяться первые производные, они являются направленными операторами. Напротив, лапласиан обладает тем важным достоинством, что он изотропен (инвариантен к повороту), обеспечивает одинаковый отклик на изменения яркости в любом направлении и позволяет избежать необходимости применения нескольких масок для вычисления максимального отклика в любой точке изображения.

Алгоритм Марра — Хилдрета состоит из свертки ЛГ-фильтра с исходным изображением

$$g(x, y) = [\nabla^2 G(x, y)] * f(x, y) \quad (6.12)$$

и нахождения затем точек пересечения нулевого уровня функции $g(x, y)$, чтобы локализовать контуры на изображении $f(x, y)$. Поскольку взятие второй производной является линейной операцией, уравнение (6.12) можно переписать в виде

$$g(x, y) = \nabla^2 [G(x, y) * f(x, y)] \quad (6.13)$$

как если бы изображение сначала сворачивалось с гауссовой сглаживающей функцией, а потом вычислялся лапласиан результата. Оба последних уравнения дают идентичные результаты.

Алгоритм выделения контуров Марра — Хилдрета можно кратко суммировать следующим образом:

- 1) исходное изображение обрабатывается гауссовым фильтром низких частот с размерами $n \times n$, полученным посредством дискретизации уравнения (6.9);
- 2) вычисляется лапласиан полученного изображения, используя, например, маску (6.11). (Первый и второй шаги вместе реализуют уравнение (6.13));
- 3) на полученном изображении находятся точки пересечения нулевого уровня.

Детектор контуров Кэнни

Детектор контуров Кэнни [104] использует более сложный алгоритм, но зато качество его работы в общем случае выше, чем у рассмотренных раньше алгоритмов выделения контуров. Метод Кэнни преследует три основные цели:

- 1) *низкая частота ошибок*. Должны обнаруживаться все контуры с минимумом ложных срабатываний. Иначе говоря, обнаруженный контурный рисунок должен быть как можно ближе к истинному;
- 2) *хорошая локализация контурных точек*. Найденное пространственное положение контуров должно быть как можно ближе к истинному, т. е. расстояние от точки, которую детектор опознает как точку контура, до середины истинного контура должно быть минимальным;
- 3) *одинокый отклик на точку контура*. Для каждой точки истинного контура детектор должен обнаруживать только одну точку. Другими словами, количество локальных максимумов в окрестности истинной контурной точки должно быть минимальным. Это значит, что детектор не должен обнаруживать несколько контурных пикселей, если имеется контур, состоящий только из одной точки.

Алгоритм выделения контуров Кэнни состоит из следующих основных шагов:

- 1) сгладить исходное изображение гауссовым фильтром;
- 2) сформировать изображения модуля и направления градиента;
- 3) применить подавление немаксимальных точек к изображению модуля градиента;
- 4) выполнить преобразование с двойным порогом и анализ связности для обнаружения и связывания контуров.

Подробное описание метода Кэнни можно найти в [98].

Дискретное преобразование Фурье и построение спектра

Как видно из материала следующих пунктов, важную роль при решении задач фильтрации шумов и задач восстановления изображения играет обработка изображения в частотной области. Для перехода из пространственной области определения изображения в частотную

область и обратно используется дискретное преобразование Фурье, которое здесь кратко описывается. Более подробное описание можно найти в [98; 105].

Прямое Фурье-преобразование $F(u)$ непрерывной функции одной переменной $f(x)$ определяется равенством:

$$F(u) = \int_{-\infty}^{+\infty} f(x) e^{-i2\pi ux} dx.$$

Здесь переменная u играет роль нормированной частоты, имеющей размерность, обратную размерности пространственной переменной x . Из равенства видно, что элементы Фурье-образа являются комплексными величинами. Это преобразование можно распространить на функцию двух переменных следующим образом:

$$F(u, v) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x, y) e^{-i2\pi(ux+vy)} dx dy.$$

Дискретное преобразование функции одной переменной $f(x)$, $x = 0, 1, \dots, M-1$ задается равенством:

$$F(u) = \sum_{x=0}^{M-1} f(x) e^{-i2\pi ux/M}, \quad u = 0, 1, 2, \dots, M-1.$$

Вычисление Фурье-преобразования $F(u)$ осуществляется следующим образом. Вначале подставляется значение $u = 0$ в показатель экспоненты и производится суммирование по всем значениям переменной x . И далее этот процесс повторяется для всех M значений переменной u . Как и исходная функция $f(x)$, Фурье-образ является дискретной величиной и содержит то же число элементов. В отличие от непрерывного случая, дискретное преобразование Фурье и его обращение всегда существуют. Каждый из M элементов функции $F(u)$ называется частотной компонентой преобразования.

Дискретное преобразование Фурье (ДПФ) функции двух переменных $f(x, y)$, $x = 0, 1, \dots, M-1$; $y = 0, 1, \dots, N-1$ задается равенством:

$$F(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-i2\pi\left(\frac{ux}{M} + \frac{vy}{N}\right)}, \quad u = \overline{0, M-1}; v = \overline{0, N-1}. \quad (6.14)$$

Если имеется преобразование $F(u, v)$, можно получить $f(x, y)$ при помощи *обратного дискретного преобразования Фурье* (обратного ДПФ)

$$f(x, y) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} F(u, v) e^{i2\pi\left(\frac{ux}{M} + \frac{vy}{N}\right)}, x = \overline{0, M-1}; y = \overline{0, N-1}. \quad (6.15)$$

Выражения (6.14) и (6.15) составляют *пару двумерных дискретных преобразований Фурье*. Решения задач фильтрации изображений в частотной области базируются на свойствах этих двух выражений.

В заключение следует отметить взаимосвязи пространственных и частотных интервалов. Предположим, что непрерывная функция $f(t, z)$ дискретизована в виде цифрового изображения $f(x, y)$, состоящего из $M \times N$ отсчетов, взятых, соответственно, вдоль направлений t и z .

Пусть ΔT и ΔZ обозначают шаги между отсчетами в пространственной области. Тогда соответствующие дискретные интервалы в частотной области составят

$$\Delta u = \frac{1}{M\Delta T} \text{ и } \Delta v = \frac{1}{N\Delta T}$$

соответственно. Заметим, что интервалы между отсчетами в частотной области обратно пропорциональны как шагам пространственной дискретизации, так и общему числу отсчетов.

Основы фильтрации в частотной области

Фильтрация в частотной области заключается в модифицировании Фурье-образа изображения и последующем выполнении обратного преобразования для получения обработанного результата. Это утверждение можно сформулировать в виде следующего основного уравнения частотной фильтрации:

$$g(x, y) = \mathbb{F}^{-1} [H(u, v) F(u, v)], \quad (6.16)$$

где $F(u, v)$ — ДПФ исходного изображения $f(x, y)$ размерами $M \times N$, $H(u, v)$ — *фильтр-функция* (также называемая просто *фильтром*, или *передаточной функцией фильтра*), $\mathbb{F}^{-1}$ — обратное ДПФ, а $g(x, y)$ — результат фильтрации (выходное изображение). Функции F , H , и g представляют собой массивы размерами $M \times N$, как и исходное изображение. Произведение $H(u, v)F(u, v)$ формируется как поэлементное произведение массивов. Фильтр-функция модифицирует Фурье-преобразование исходного изображения, чтобы получить на выходе обработанное изображение $g(x, y)$. Задание функции $H(u, v)$ значитель-

но упростится, если использовать функции, симметричные относительно своего центра; это в свою очередь требует, чтобы $F(u, v)$ также была центрирована тем же самым образом. Как объясняется в [98], это достигается умножением исходного изображения на $(-1)^{x+u}$ перед выполнением прямого ДПФ.

Если H является действительной и симметричной, а f — действительной (как это обычно и бывает в реальной ситуации), то обратное ДПФ в (6.16) теоретически должно в результате давать действительные значения. Однако на практике обратное преобразование обычно имеет паразитные комплексные члены из-за округлений и других неточностей вычислений. Поэтому для формирования g обычно берут действительную часть обратного ДПФ.

Соответствие между пространственными и частотными фильтрами

Связь между фильтрами пространственной и частотной областей определяется теоремой о свертке [98], которая утверждает, что преобразование Фурье свертки двух функций равно произведению спектров этих функций.

Пусть задан фильтр $H(u, v)$, необходимо определить его эквивалентное представление в пространственной области формулой пространственной свертки (6.2). Для этого используется единичный импульс $f(x, y) = \delta(x, y)$. Из свойств ДПФ (см. формулу (6.14)) легко следует, что $F(u, v) = 1$. Тогда выход фильтра будет равен обратному преобразованию частотного фильтра, соответствующего пространственному фильтру. Из аналогичного анализа и теоремы о свертке следует и обратное, что если задан пространственный фильтр, его представление в частотной области получается выполнением прямого Фурье-преобразования пространственного фильтра. Таким образом, два фильтра образуют Фурье-пару:

$$h(x, y) \Leftrightarrow H(u, v), \quad (6.17)$$

где $h(x, y)$ — пространственный фильтр. Поскольку этот фильтр может быть получен как отклик частотного фильтра на единичный импульс, то часто $h(x, y)$ называют *импульсной характеристикой* (импульсным откликом) фильтра $H(u, v)$. Также, поскольку при дискретной реализации уравнения (6.17) как пространственные (временные) параме-

тры, так и значения коэффициентов являются ограниченными, такие фильтры называют фильтрами *с конечными импульсными характеристиками* (КИХ-фильтрами).

Пространственная свертка была рассмотрена ранее, и ее реализация рассматривалась в связи с уравнением (6.2), которое относится к свертке функций различных размеров. Когда говорится о пространственной свертке в терминах теоремы о свертке и ДПФ, то подразумевается, что в операции участвуют периодические функции. Более того, свертка в контексте ДПФ предполагает наличие одинаковых по размерам изображений, тогда как в (6.2) функции могут быть и совершенно различных размеров.

На практике часто требуется осуществлять фильтрацию сверткой (согласно уравнению (6.2)) с маской небольшого размера, что может выполняться с достаточно высокой скоростью и не слишком трудно в аппаратной реализации. Однако идеи фильтрации в частотной области интуитивно более понятны. Один из способов воспользоваться преимущественными свойствами каждой из областей состоит в том, чтобы задать фильтр в частотной области, вычислить его обратное ДПФ и полученный полноразмерный пространственный фильтр использовать в качестве образца для построения масок пространственных фильтров меньшего размера. Возможно рассмотреть и обратный случай, когда задан небольшой пространственный фильтр и формируется его полноразмерное представление в частотной области. Такой подход удобен для частотного анализа поведения пространственных фильтров.

В качестве примера того, как частотные фильтры могут использоваться в качестве эталона для задания коэффициентов некоторых из небольших масок, рассматривавшихся выше, воспользуемся гауссовым фильтром. Фильтры, основанные на гауссовых функциях, представляют особый интерес, поскольку и прямое, и обратное Фурье-преобразования гауссовой функции являются действительными гауссовыми функциями. Мы ограничимся рассмотрением одномерного случая для иллюстрации лежащих в основе фильтрации принципов. Двумерные гауссовы фильтры будут рассмотрены позже.

Обозначим $H(u)$ одномерный частотный гауссов фильтр:

$$H(u) = Ae^{-u^2/2\sigma^2},$$

где σ — стандартное отклонение гауссовой функции. Для получения соответствующего пространственного фильтра нужно взять обратное преобразование Фурье от $H(u)$:

$$h(x) = \sqrt{2\pi\sigma} A e^{-\pi^2\sigma^2 x^2}.$$

Данные выражения важны по двум причинам:

- 1) они образуют Фурье-пару, причем каждая функция является гауссовой и *действительной*. Это облегчает анализ, поскольку нет необходимости заниматься комплексными числами. Кроме того, гауссовы кривые наглядны и удобны для обработки;
- 2) функции ведут себя взаимно противоположно: если $H(u)$ широкая (значение σ большое), то $h(x)$ узкая, и наоборот. Фактически, если σ устремить в бесконечность, то $H(u)$ будет стремиться к константе, а $h(x)$ — к импульсу. То есть в пределе это означает отсутствие фильтрации.

Практическое задание к п. 6.1

Выполните базовые операции по обработке изображений лица с веб-камеры:

- 1) реализуйте алгоритм эквализации гистограммы;
- 2) реализуйте пороговую обработку изображения;
- 3) реализуйте алгоритмы выделения контуров;
- 4) постройте спектр изображения.

6.2. Методы улучшения изображений

В этом пункте описываются методы обработки изображений в пространственной и частотных областях для задач фильтрации шумов и повышения резкости.

Как правило, шумы на изображении попадают в области высоких частот, поэтому борьба с ними осуществляется с помощью низкочастотных фильтров, которым в пространственной области соответствуют сглаживающие фильтры.

Сглаживание изображений

Сглаживающие фильтры применяются для расфокусировки изображения и подавления шума. Расфокусировка может применяться в задачах предварительной обработки изображения, например для удаления мелких деталей перед обнаружением больших объектов или же для устранения разрывов в линиях или деталях. Для подавления шумов может использоваться расфокусировка с применением как линейной, так и нелинейной фильтрации.

Линейные сглаживающие фильтры

Выход (отклик) простейшего линейного сглаживающего пространственного фильтра есть среднее значение элементов по окрестности, покрытой маской фильтра. Такие фильтры иногда называют *усредняющими*, или *сглаживающими*, фильтрами. Как упоминалось выше, их также называют *низкочастотными* фильтрами.

Как следует из уравнения (6.1), общая формула фильтрации изображения размерами $M \times N$ фильтром взвешенного среднего по окрестности $m \times n$ (m и n — нечетные) задается выражением

$$g(x, y) = \frac{\sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x + s, y + t)}{\sum_{s=-a}^a \sum_{t=-b}^b w(s, t)}. \quad (6.18)$$

Параметры в этом уравнении такие же, как были определены для уравнения (6.1). Как и ранее, подразумевается, что полная фильтрация изображения достигается применением формулы (6.18) ко всем парам $x = 0, 1, 2, \dots, M - 1$ и $y = 0, 1, 2, \dots, N - 1$. Знаменатель в (6.18) есть просто сумма всех коэффициентов маски, следовательно, является константой и требует вычисления лишь один раз.

Далее будет рассмотрен метод сглаживания в частотной области, который осуществляется с помощью низкочастотных фильтров. Значительный вклад в высокие частоты Фурье-преобразования изображения дают контуры и другие резкие яркостные переходы, включая шумы. Следовательно, в частотной области сглаживание (размывание) достигается подавлением высоких частот, т. е. при помощи *низкочастотной* фильтрации. Ниже будут рассмотрены два вида низкочастотных фильтров: фильтр Баттерворта и гауссов фильтр. Фильтр

Баттерворта характеризуется параметром, который называется *порядком* фильтра. При больших значениях этого параметра фильтр Баттерворта приближается по форме к идеальному фильтру. При малых значениях он имеет гладкую форму, похожую на форму гауссова фильтра. Таким образом, фильтр Баттерворта может рассматриваться как переходный между двумя «крайностями».

Далее предполагается, что все фильтр-функции $H(u, v)$ являются дискретными функциями размеров $P \times Q$, т. е. дискретные частотные переменные находятся в диапазонах $u = 0, 1, 2, \dots, P - 1$ и $v = 0, 1, 2, \dots, Q - 1$.

Фильтры низких частот Баттерворта

Передаточная функция низкочастотного фильтра Баттерворта (ФНЧ Баттерворта) порядка n с частотой среза на расстоянии D_0 от начала координат задается формулой

$$H(u, v) = \frac{1}{1 + [D(u, v) / D_0]^{2n}}, \quad (6.19)$$

где расстояние $D(u, v)$ задано формулой

$$D(u, v) = \sqrt{\left(u - \frac{P}{2}\right)^2 + \left(v - \frac{Q}{2}\right)^2}. \quad (6.20)$$

Трехмерный график, полутоновое изображение и радиальные профили передаточной функции ФНЧ Баттерворта представлены на рис. 6.5.

Передаточная функция ФНЧ Баттерворта обычно записывается как квадратный корень из выражения (6.19). Однако здесь представляет интерес в основном форма фильтра и поэтому исключается квадратный корень для удобства.

Передаточная функция ФНЧ Баттерворта не имеет разрыва, который задает точную границу между пропускаемыми и обрезаемыми частотами. Для фильтров с гладкой передаточной функцией обычной практикой является определение местоположения частот среза как множества точек, в которых значения функции $H(u, v)$ становятся меньше определенной доли ее максимального значения. В выражении (6.19) такой точкой является $D(u, v) = D_0$, когда $H(u, v)$ становится меньше 50 % максимального значения, равного 1.

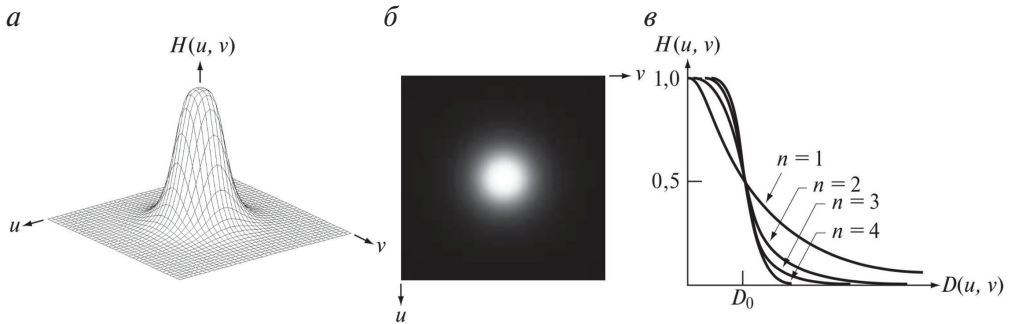


Рис. 6.5. Фильтр низких частот Баттерворта:

a — трехмерный график передаточной функции низкочастотного фильтра Баттерворта;
б — полутоновое изображение фильтра; *в* — радиальные профили фильтров с порядками от 1 до 4

Гауссовы фильтры низких частот

Для одномерного случая гауссовы фильтры низких частот (ФНЧ Гаусса) были введены в конце предыдущего пункта 6.1, где они использовались для того, чтобы установить некоторые важные взаимосвязи между пространственной и частотной областями. В двумерном случае эти фильтры задаются формулой

$$H(u, v) = e^{-D^2(u, v)/2\sigma^2}, \quad (6.21)$$

где, как и в (6.20), $D(u, v)$ — расстояние от центра частотного прямоугольника. Здесь опускается константа перед выражением, задающим фильтр, чтобы сохранить единообразие с остальными фильтрами, рассматриваемыми в настоящей главе, максимальное значение которых равно 1.

Как и раньше, σ задает размах вокруг центра. Обозначив $\sigma = D_0$, мы можем переписать выражение для фильтра, используя форму записи остальных фильтров данного пункта:

$$H(u, v) = e^{-D^2(u, v)/2D_0^2}, \quad (6.22)$$

где D_0 — частота среза. Когда $D(u, v) = D_0$, значение передаточной функции ФНЧ Гаусса падает до 0,607 от своего максимального значения.

Как известно, обратное Фурье-преобразование от ФНЧ Гаусса также есть гауссова функция. Это означает, что пространственный гауссов фильтр, полученный вычислением обратного ДПФ согласно вы-

ражениям (6.21) или (6.22), не будет создавать паразитных колебаний. Трехмерный график ФНЧ Гаусса, его полутонное изображение и радиальные профили передаточной функции ФНЧ Гаусса представлены на рис. 6.6.

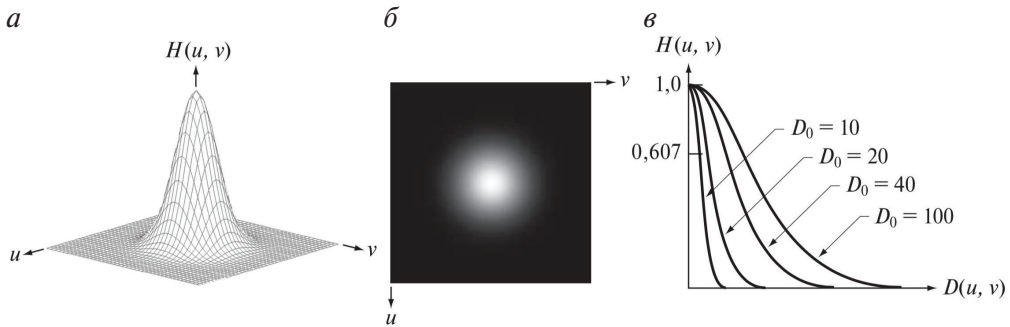


Рис. 6.6. Гауссов фильтр низких частот:

a — трехмерный график передаточной функции ФНЧ Гаусса; *б* — полутонное изображение фильтра; *в* — радиальные профили фильтров для различных значений D_0

Фильтры, основанные на порядковых статистиках (нелинейные фильтры)

Фильтры, основанные на порядковых статистиках, относятся к классу нелинейных пространственных фильтров. Отклик такого фильтра определяется предварительным упорядочиванием (ранжированием) значений пикселей, покрываемых маской фильтра, и последующим выбором значения, находящегося на определенной позиции упорядоченной последовательности (т. е. имеющего определенный ранг). Собственно фильтрация сводится к замещению исходного значения пикселя (в центре маски) на полученное значение отклика фильтра. Наиболее известен медианный фильтр, который, как следует из названия, заменяет значение пикселя на значение медианы распределения яркостей всех пикселей в окрестности (включая и исходный). Медианные фильтры весьма популярны потому, что для определенных типов случайных шумов они демонстрируют отличные возможности подавления шума при значительно меньшем эффекте расфокусировки, чем у линейных сглаживающих фильтров с аналогичными размерами. В частности, медианные фильтры эффективны при фильтрации импульсных шумов, иногда называемых шумами «соль и перец», которые выглядят как наложение на изображение случайных черных и белых точек.

Медиана набора чисел есть такое число ξ , что половина чисел из набора меньше или равна ξ , а другая половина — больше или равна ξ . Чтобы выполнить медианную фильтрацию для элемента изображения, необходимо сначала упорядочить по возрастанию значения пикселей внутри окрестности, затем найти значение медианы и, наконец, присвоить полученное значение обрабатываемому элементу.

Повышение резкости изображений

Главная цель повышения резкости заключается в том, чтобы подчеркнуть яркостные переходы. Повышение резкости изображений используется достаточно широко — от цифровой печати и медицинской интроскопии до технического контроля в промышленности и систем автоматического наведения в военной сфере. В предыдущих подразделах можно было видеть, что расфокусировка изображения может быть достигнута пространственной операцией усреднения значений точек по окрестности. Поскольку усреднение аналогично интегрированию, то логично прийти к выводу, что повышение резкости, будучи явлением, обратным по отношению к расфокусировке, может быть достигнуто пространственным дифференцированием.

Повышения резкости в пространственной области

Рассмотрим применение двумерной второй производной в задачах повышения резкости изображений. Наиболее полезны для этой задачи изотропные фильтры, отклик которых не зависит от направления неоднородностей на обрабатываемом изображении. Другими словами, изотропные фильтры являются инвариантными к повороту в том смысле, что поворот изображения и последующее применение фильтра дает тот же результат, что и первоначальное применение фильтра с последующим поворотом результата. Можно показать, что простейшим изотропным оператором, основанным на производных, является лапласиан (оператор Лапласа), который в случае функции двух переменных $f(x, y)$ определяется как

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

Дискретная реализация оператора Лапласа может быть реализована с помощью свертки изображения с окном (6.11).

Поскольку оператор Лапласа по сути является второй производной, его применение подчеркивает разрывы уровней яркостей на изображении и подавляет области со слабыми изменениями яркостей. Это приводит к получению изображения, содержащего сероватые линии на месте контуров и других разрывов, наложенные на темный фон без особенностей. Но фон можно «восстановить», сохранив при этом эффект повышения резкости, достигаемый лапласианом. Для этого достаточно вычесть изображение-лапласиан из исходного изображения. Вычитание требуется благодаря отрицательному коэффициенту окна в маске лапласиана (6.11). Таким образом, алгоритм использования лапласиана для повышения резкости изображений сводится к следующему:

$$g(x, y) = f(x, y) - [\nabla^2 f(x, y)], \quad (6.23)$$

где $f(x, y)$ и $g(x, y)$ — исходное изображение и изображение с повышенной резкостью.

Повышения резкости изображений частотными фильтрами

В предыдущем пункте было показано, что изображение может быть сглажено путем подавления высокочастотных составляющих его Фурье-преобразования. Поскольку контуры и другие скачкообразные изменения яркости связаны с высокочастотными составляющими, повышение резкости изображения может быть достигнуто при помощи процедуры высокочастотной фильтрации в частотной области, которая, наоборот, подавляет низкочастотные составляющие и не затрагивает высокочастотную часть Фурье-преобразования. Как и ранее, мы рассматриваем центрально-симметричные фильтры с нулевым фазовым сдвигом.

Здесь, как и ранее, все фильтр-функции $H(u, v)$ понимаются как дискретные функции размерами $P \times Q$; т. е. дискретные частотные переменные находятся в диапазоне $u = 0, 1, 2, \dots, P - 1$ и $v = 0, 1, 2, \dots, Q - 1$.

Передаточная функция высокочастотного фильтра может быть получена из заданного низкочастотного фильтра при помощи соотношения

$$H_{HP}(u, v) = 1 - H_{LP}(u, v). \quad (6.24)$$

Фильтры высоких частот Баттерворта

Передаточная функция двумерного *высокочастотного фильтра Баттерворта* (ФВЧ Баттерворта) порядка n с частотой среза D_0 задается формулой

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

где расстояние $D(u, v)$ вычисляется согласно (6.20). Это выражение прямо следует из (6.19) и (6.24). На рис. 6.7 в среднем ряду представлено изображение и профиль передаточной функции ФВЧ Баттерворта.

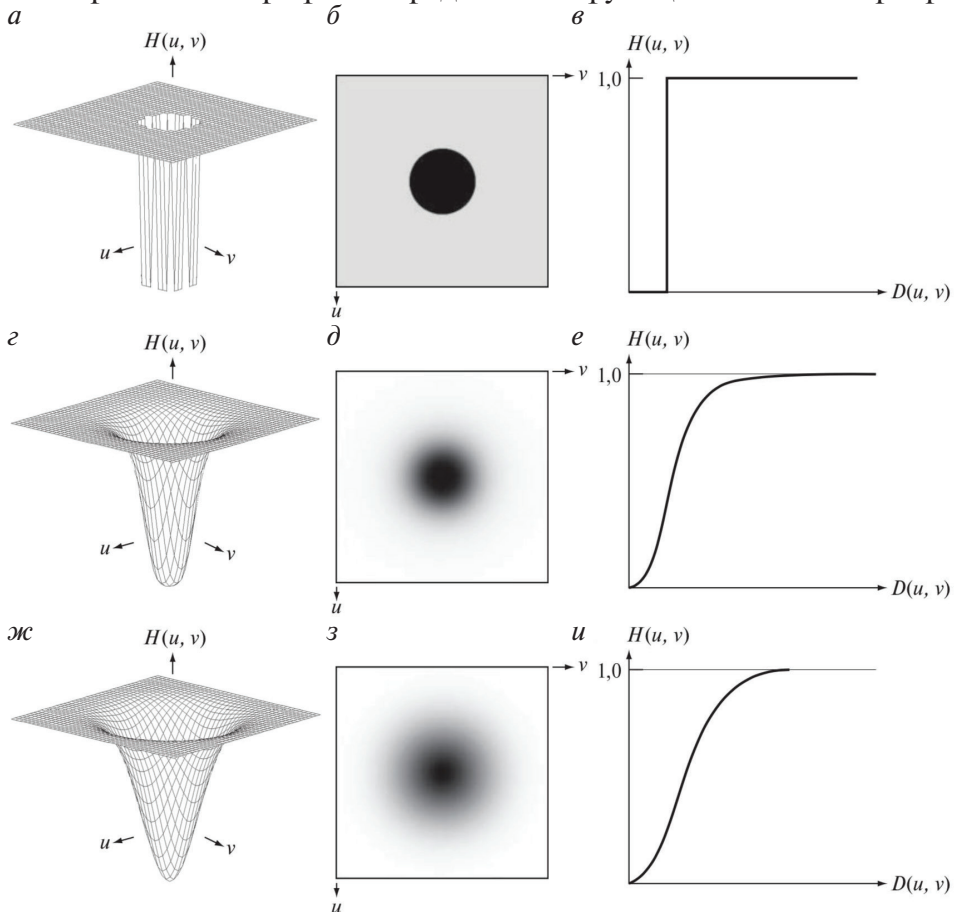


Рис. 6.7. Частотные фильтры для повышения резкости изображений:
 верхний ряд: трехмерный график, полутоновое изображение и профиль типичного
 идеального высокочастотного фильтра; средний и нижний ряды: та же последовательность
 для типичных высокочастотных фильтров Баттерворта и Гаусса

Гауссовы фильтры высоких частот

Передаточная функция гауссова фильтра высоких частот (ФВЧ Гаусса) с частотой среза, расположенной на расстоянии D_0 от центра частотного прямоугольника, задается формулой

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

где расстояние $D(u, v)$ вычисляется согласно (6.20). Это выражение прямо следует из (6.22) и (6.24). На рис. 6.7 в нижнем ряду представлены трехмерный график, изображение и профиль передаточной функции ФНЧ Гаусса.

Лапласиан в частотной области

Выше в этом пункте лапласиан в пространственной области уже применялся для улучшения изображения. Здесь мы вернемся к лапласиану еще раз и покажем, что при использовании частотных методов он дает эквивалентные результаты. Можно показать, что лапласиан может быть реализован в частотной области при помощи фильтра

$$H(u, v) = -4\pi^2(u^2 + v^2).$$

Используя функцию расстояния от центра частотной области $D(u, v)$, определенную выражением (6.20), можно переписать передаточную функцию фильтра в виде

$$H(u, v) = -4\pi^2 \left[\left(u - \frac{P}{2} \right)^2 + \left(v - \frac{Q}{2} \right)^2 \right] = -4\pi^2 D^2(u, v).$$

Тогда лапласиан изображения будет выглядеть как

$$\nabla^2 f(x, y) = \mathbb{F}^{-1} [H(u, v) F(u, v)], \quad (6.25)$$

где $F(u, v)$ есть ДПФ от $f(x, y)$.

Как показано ранее, повышение резкости изображения в пространственной области достигается использованием выражения (6.23). При этом функции $f(x, y)$ и $\nabla^2 f(x, y)$ тогда имели сравнимые по величине значения. Однако вычисление $\nabla^2 f(x, y)$ через передаточную функцию согласно выражению (6.25) вносит масштабный коэффициент обрат-

ного ДПФ, который может на несколько порядков превышать максимальное значение f . Значит, необходимо вернуть разницу между f и ее лапласианом в сравнимые диапазоны. Простейший путь решения этой проблемы состоит в нормировке значений $f(x, y)$ в диапазон $[0, 1]$ перед вычислением ДПФ и делением лапласиана $\nabla^2 f(x, y)$ на свое максимальное значение, что приведет его в приблизительный диапазон $[-1, 1]$ (напомним, что лапласиан имеет также и отрицательные значения). После этого уже можно применять выражение (6.23).

Практическое задание к п. 6.2

Выполните операции для улучшения изображений лица с веб-камеры:

- 1) реализуйте низкочастотную фильтрацию изображений;
- 2) реализуйте один из методов повышения резкости.

6.3. Восстановление изображений

Основной целью операции восстановления является реконструкция изображения, то есть преодоление последствий действия определенных искажающих факторов, таких, как шум, расфокусировка или смаз. При восстановлении делается попытка реконструировать или воссоздать изображение, используя априорную информацию о явлении, которое вызвало ухудшение изображения. Поэтому методы восстановления основаны на моделировании процессов искажения и применении обратных математических процедур для воссоздания исходного изображения. Изложение материала данного пункта, в основном, следует монографии [98].

Модель процесса искажения изображения

На рис. 6.8 показана принятая в этой главе модель процесса искажения, которая предполагает действие некоторого искажающего оператора H на исходное изображение $f(x, y)$, что после добавления

аддитивного шума дает искаженное изображение $g(x, y)$. Задача восстановления состоит в построении некоторого приближения $\hat{f}(x, y)$ исходного изображения по заданному (искаженному) изображению $g(x, y)$, некоторой информации относительно искажающего оператора H и некоторой информации относительно аддитивного шума $\eta(x, y)$. Необходимо, чтобы это приближение было как можно ближе к исходному изображению, и, в принципе, чем больше сведений об операторе H и о функции η , тем ближе будет функция $\hat{f}(x, y)$ к функции $f(x, y)$. В основе подхода лежит использование операторов (фильтров), восстанавливающих изображение.

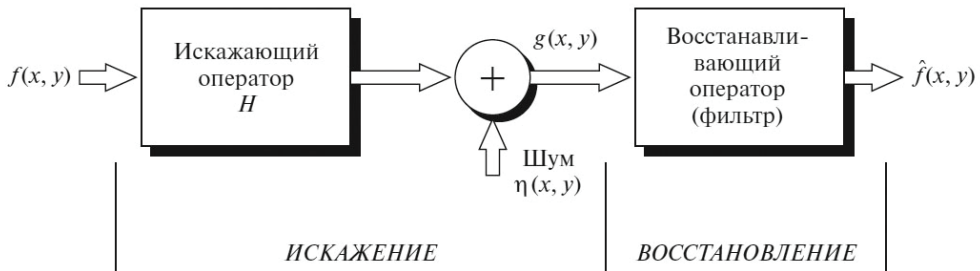


Рис. 6.8. Модель процесса искажения/восстановления изображения [98]

Известно, что если H — линейный трансляционно инвариантный оператор, то искаженное изображение может быть представлено в пространственной области в виде:

$$g(x, y) = h(x, y) * f(x, y) + \eta(x, y), \quad (6.26)$$

где $h(x, y)$ — функция, представляющая искажающий оператор в пространственной области, а символ «*», как обычно, используется для обозначения пространственной свертки. Из материала предыдущей главы нам известно, что свертка в пространственной области аналогична умножению в частотной области, поэтому равенство (6.26) может быть эквивалентным образом записано в частотной области:

$$G(u, v) = H(u, v)F(u, v) + N(u, v), \quad (6.27)$$

где обозначенные заглавными буквами функции есть Фурье-образы соответствующих функций в (6.26), а умножение понимается как поэлементное. Так, элемент произведения $H(u, v)F(u, v)$ с номером ij есть произведение ij -того элемента $H(u, v)$ на ij -тый элемент $F(u, v)$.

Равенства (6.26) и (6.27) составляют основу линейных методов восстановления изображений.

Для дальнейшего рассмотрения важными являются параметры, определяющие пространственные характеристики шума, а также вопрос, коррелирует ли шум с изображением. Под частотными характеристиками понимаются свойства спектра шума в смысле преобразования Фурье. Например, шум, спектр которого является постоянной величиной, называется обычно *белым шумом*. Происхождение этого термина связано с физическими свойствами белого света, который содержит практически все частоты видимого спектра в равных пропорциях.

В рамках сделанных выше предположений мы будем иметь дело с описанием поведения шума в *пространственной области*, которое основано на статистических свойствах значений интенсивности компоненты шума в модели на рис. 6.8. Эти значения яркости могут рассматриваться как случайные величины, характеризующиеся функцией плотности распределения вероятностей. Ниже даны примеры функций, которые наиболее часто встречаются в приложениях, связанных с обработкой изображений.

Гауссов шум

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

Функция плотности распределения вероятностей гауссовой скалярной случайной величины z задается выражением

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(z-\bar{z})^2}{\sigma^2}}, \quad (6.28)$$

где z представляет собой значение яркости, — среднее значение случайной величины z , σ — ее среднеквадратическое отклонение. Квадрат среднеквадратического отклонения σ^2 называется *дисперсией* величины z . График этой функции представлен на рис. 6.9, а. Когда плотность распределения случайной величины z описывается функцией (6.28), то приблизительно 70 % ее значений попадают в диапазон $(\bar{z} - \sigma, \bar{z} + \sigma)$ и примерно 95 % — в диапазон $(\bar{z} - 2\sigma, \bar{z} + 2\sigma)$.

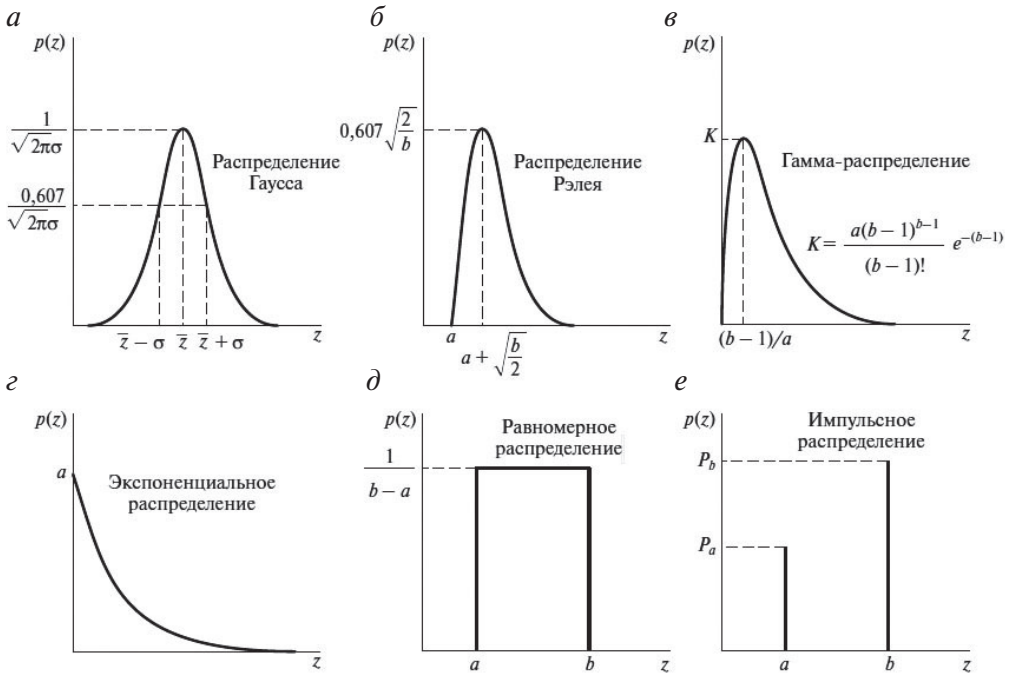


Рис. 6.9. Некоторые важные функции плотности распределения вероятностей [98, с. 370]

Шум Рэлея

Функция плотности распределения вероятностей шума Рэлея задается выражением

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

Среднее и дисперсия для этого распределения имеют вид

$$\bar{z} = a + \sqrt{\pi b / 4} \quad \text{и} \quad \sigma^2 = \frac{b(4 - \pi)}{4}.$$

График плотности распределения вероятностей шума Рэлея представлен на рис. 6.9, б. Необходимо обратить внимание на местоположение начала координат и на то обстоятельство, что график имеет асимметричную (перекошенную вправо) форму. Распределение Рэлея бывает полезно для приближения асимметричных гистограмм.

Шум Эрланга (гамма-шум)

Функция плотности распределения вероятностей шума Эрланга задается выражением

$$p(z) = \begin{cases} \frac{a^b z^{b-1}}{(b-1)!} e^{-az}, & \text{при } z \geq a, \\ 0, & \text{при } z < a, \end{cases} \quad (6.29)$$

где $a > 0$, b — положительное целое число и символ «!» обозначает факториал.

Среднее и дисперсия для этого распределения имеют вид

$$\bar{z} = \frac{b}{a} \text{ и } \sigma^2 = \frac{b}{a^2}.$$

На рис. 6.9, в представлен график плотности этого распределения. Выражение (6.29) часто называют *гамма-распределением*, хотя, строго говоря, это название относится к распределению более общего вида, когда b не является целым, а в знаменателе стоит гамма-функция $\Gamma(b)$. Рассматриваемый частный случай правильнее называть *распределением Эрланга*.

Экспоненциальный шум

Частным случаем распределения Эрланга при $b = 1$ является экспоненциальное распределение, функция плотности которого описывает так называемый *экспоненциальный шум* и задается выражением:

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

где $a > 0$. Среднее и дисперсия для этого распределения имеют вид

$$\bar{z} = \frac{1}{a} \text{ и } \sigma^2 = \frac{1}{a^2}.$$

Равномерный шум

Функция плотности распределения вероятностей *равномерного* шума задается выражением

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

Среднее значение для этого распределения равно

$$\bar{z} = \frac{a+b}{2},$$

а дисперсия

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

На рис. 6.9, д представлен график плотности этого распределения.

Импульсный шум

Функция плотности распределения вероятностей (*биполярного*) импульсного шума задается выражением

$$p(z) = \begin{cases} P_a & \text{при } z = a; \\ P_b & \text{при } z = b; \\ 1 - P_a - P_b, & \text{иначе.} \end{cases}$$

Если $b > a$, то пиксель с яркостью b выглядит как светлая точка на изображении. Пиксель с яркостью a выглядит, наоборот, как темная точка. Если одно из значений вероятности (P_a или P_b) равно нулю, то импульсный шум называется *униполярным*. Если ни одна из вероятностей не равна нулю и в особенности если они приблизительно равны по величине, импульсный шум походит на крупичи соли и перца, случайно рассыпанные по изображению. По этой причине импульсный шум называют также шумом типа «*соль и перец*».

Значения импульсов шума могут быть как положительные, так и отрицательные. При оцифровке изображения обычно происходит масштабирование (и ограничение) значений яркости. Поскольку величина связанных с импульсным шумом искажений, как правило, велика по сравнению с величиной полезного сигнала, импульсный шум после оцифровки принимает экстремальные значения, что соответствует появлению абсолютно черных и белых точек на изображении. Поэтому обычно предполагается, что значения a и b являются «интенсивными» в том смысле, что они равны минимальному и максимальному

значениям, которые в принципе могут присутствовать в оцифрованном изображении. В результате отрицательные импульсы выглядят как черные точки на изображении (перец). По тем же причинам положительные импульсы выглядят как белые точки (соль). Для 8-битовых изображений это обычно означает, что $a = 0$ (черное) и $b = 255$ (белое). На рис. 6.9, *e* представлен график функции плотности вероятностей значений импульсного шума.

Рассмотренные распределения в совокупности представляют собой набор средств, которые позволяют моделировать искажения, связанные с широким диапазоном встречающихся на практике шумов. Так, например, шум сенсоров видеокамер, возникающий из-за недостатка освещения и/или высокой температуры, описывается гауссовым шумом. Распределение Рэлея полезно при моделировании шума, который возникает на снимках, снятых с большого расстояния. Экспоненциальное и гамма-распределения отвечают шуму на изображениях, получаемых с использованием лазеров. С импульсным шумом можно столкнуться в ситуациях, когда во время оцифровки изображения из-за помех в сети питания возникают переходные процессы, приводящие к появлению экстремальных значений, о которых говорилось в предыдущем абзаце. Равномерное распределение в наименьшей степени подходит для описания встречающихся на практике явлений. Однако это распределение весьма полезно как основа для создания различных генераторов случайных чисел, используемых при моделировании.

Когда искажение изображения обусловлено исключительно наличием шума, равенства (6.26) и (6.27) приобретают вид

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

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

Слагаемое, описывающее шум, неизвестно, поэтому просто вычесть его из функции $g(x, y)$ или $G(u, v)$ невозможно. В тех ситуациях, когда на изображении присутствует только аддитивный случайный шум, пространственная фильтрация является лучшим из возможных методов восстановления.

Линейные трансляционно-инвариантные модели (6.26) и (6.27) могут быть использованы для приближенного описания многих типов искажений. Преимущество такого подхода заключается в том, что огромное количество используемых в линейной теории методов и средств

становится применимо для решения задач восстановления изображений. Хотя нелинейные и трансляционно-неинвариантные методы являются более общими (и обычно более точными), но их использование часто приводит к непреодолимым или очень трудно решаемым численными методами проблемам. Описываемый здесь материал касается рассмотрения линейных трансляционно-инвариантных методов. Поскольку искажение представляет собой результат свертки, то для восстановления необходимо найти такой фильтр, применение которого приводило бы к обратному процессу. Поэтому для обозначения линейного процесса восстановления часто используется термин *реконструкция (деконволюция) изображений*. В связи с этим фильтры, используемые для восстановления, часто называются *реконструирующими фильтрами*.

Оценка искажающей функции

Существуют три основных способа оценки искажающей функции $h(x, y)$ (6.26) (ядра искажающего оператора) для последующего ее использования при восстановлении изображений: 1) визуальный анализ; 2) эксперимент и 3) математическое моделирование. Вследствие того, что истинная искажающая функция нечасто бывает известна полностью, процесс восстановления изображения с использованием приближения искажающей функции, полученного некоторым образом, иногда называют реконструкцией «вслепую».

Оценка на основе визуального анализа изображения

Предположим, что имеется искаженное изображение, но информация об искажающей функции H отсутствует. Основываясь на предположении, что искажающий изображение процесс являлся линейным и пространственно-инвариантным, один из способов оценить эту функцию состоит в выделении информации непосредственно из изображения. Например, если изображение является размытым, мы можем рассмотреть его небольшой прямоугольный фрагмент, содержащий примеры структуры, такие как часть некоторого объекта и фон. Для того чтобы уменьшить влияние шума, следует выбрать ту область изображения, которая содержит полезный сигнал большой амплитуды.

ды (т. е. фрагмент высокого контраста). Следующий этап состоит в такой обработке выбранного фрагмента изображения, чтобы насколько возможно максимально убрать размытость. Например, для этого можно использовать повышение резкости с помощью соответствующих фильтров или даже обработать небольшую область вручную.

Обозначим рассматриваемую часть изображения как $g_s(x, y)$ и обработанный фрагмент (который в действительности представляет собой наше приближение для части неискаженного изображения в рассматриваемой области) как $f_s(x, y)$. Далее, предполагая, что влияние шума пренебрежимо мало в силу нашего выбора области с большим полезным сигналом, на основании (6.26) имеем

$$H_s(u, v) = \frac{G_s(u, v)}{F_s(u, v)}.$$

Исходя из свойств функции $H_s(u, v)$ и опираясь на предположение о трансляционной инвариантности искажений, теперь можно вывести полную искажающую функцию $H(u, v)$. Предположим, например, что радиальный профиль функции $H_s(u, v)$ приблизительно совпадает с формой гауссовой кривой. Это может быть использовано для построения функции $H(u, v)$ той же самой формы, но большего размера. Затем $H(u, v)$ используется одним из методов восстановления, которые будут рассмотрены в последующих пунктах. Ясно, что это является достаточно трудоемким процессом, применяющимся лишь в исключительных обстоятельствах при восстановлении уникальных изображений.

Оценка на основе эксперимента

Если оборудование, аналогичное тому, которое использовалось при получении изображения, доступно, то в принципе возможно получить точную оценку искажающей функции. Сначала необходимо так подобрать параметры регистрирующей системы, чтобы искажения на получаемых с ее помощью изображениях, похожих по сценарию на подлежащее восстановлению изображение, как можно лучше соответствовали искажениям на этом изображении. Далее идея состоит в том, чтобы сформировать импульсный отклик (ядро искажающего оператора), для чего нужно получить изображение импульса (ма-

ленькой яркой точки), используя систему с подобранными значениями параметров. Как известно, линейная трансляционно-инвариантная система определяется полностью своим импульсным откликом. Импульс симулируется яркой световой точкой. Чтобы уменьшить влияние шума, яркость должна быть как можно больше. Затем, учитывая, что Фурье-преобразование импульса есть константа, из (6.26) получаем

$$H(u, v) = \frac{G(u, v)}{A},$$

где, как и раньше, $G(u, v)$ — Фурье-преобразование полученного изображения, A — константа, описывающая величину яркости импульса.

Оценка на основе моделирования

Математическое моделирование искажений используется уже в течение многих лет, так как оно позволяет проникнуть в суть задачи восстановления изображений. В некоторых случаях модель позволяет даже учесть внешние условия, которые вызывают искажения, условия освещенности, задымленность и т. п. Важным аспектом моделирования является построение математической модели непосредственно из основных принципов формирования изображения. Например, в случае, когда размытие (смазывание) возникает в результате равномерного поступательного движения изображения сцены относительно регистрирующей системы в процессе фотосъемки, оказывается возможным [98] получить аналитическое выражение для искажающего ядра.

Фильтр Винера

В настоящей главе рассмотрен метод, соединяющий в себе учет свойств искажающей функции и статистических свойств шума в процессе восстановления. Метод основан на рассмотрении изображения и шума как случайных величин, и задача ставится следующим образом: найти такую оценку $\hat{f}$ для неискаженного изображения f , чтобы средний квадрат отклонения этих величин друг от друга (ошибка) был минимальным. Данная мера отклонения e задается формулой

$$e^2 = \left\{ (f - \hat{f})^2 \right\}, \quad (6.30)$$

где $E\{\cdot\}$ обозначает математическое ожидание своего аргумента. Предполагается, что выполнены следующие условия: 1) шум и неискаженное изображение не коррелированы между собой; 2) либо шум, либо неискаженное изображение имеют нулевое среднее значение; 3) оценка линейно зависит от искаженного изображения. При выполнении этих условий минимум среднего квадрата отклонения (6.30) достигается на функции, которая задается в частотной области выражением (6.31), причем последнее равенство имеет место в силу того, что произведение комплексного числа на комплексно-сопряженное равно квадрату модуля.

$$\begin{aligned}\hat{F}(u, v) &= \left(\frac{H^*(u, v) S_f(u, v)}{S_f(u, v) |H(u, v)|^2 + S_\eta(u, v)} \right) G(u, v) = \\ &= \left(\frac{H^*(u, v)}{|H(u, v)|^2 + S_\eta(u, v) / S_f(u, v)} \right) G(u, v) = \\ &= \left(\frac{1}{H(u, v) S_f(u, v)} \frac{|H(u, v)|^2}{|H(u, v)|^2 + S_\eta(u, v)} \right) G(u, v),\end{aligned}\tag{6.31}$$

где $H(u, v)$ — искажающая функция (ее частотное представление); $H^*(u, v)$ — комплексное сопряжение $H(u, v)$; $|H(u, v)|^2 = H^*(u, v) H(u, v)$; $S_\eta(u, v) = |N(u, v)|^2$ — энергетический спектр шума (см. уравнение (6.27)); $S_f(u, v) = |F(u, v)|^2$ — энергетический спектр неискаженного изображения; $G(u, v)$ — Фурье-преобразование искаженного изображения.

Приведенный результат был получен Н. Винером в 1942 году, и метод известен как оптимальная фильтрация по Винеру. Фильтр, представленный выражением внутри скобок, часто называют *фильтром минимального среднеквадратического отклонения*, или *винеровским фильтром*. Отметим, что, как видно из первой строки (6.31), проблема нулей в спектре искажающей функции при использовании винеровского фильтра не возникает, за исключением тех точек (u, v) , при которых знаменатель обращается в нуль.

Восстановленное изображение в пространственной области получается применением обратного преобразования Фурье к оценке $\hat{F}(u, v)$.

Отметим, что если шум равен нулю, то его энергетический спектр обращается в нуль, и винеровская фильтрация в этом случае сводится к инверсной фильтрации.

Когда мы имеем дело с белым шумом, спектр которого $|N(u, v)|^2$ является постоянной функцией, происходят соответствующие упрощения. Однако спектр неискаженного изображения редко бывает известен. В тех случаях, когда спектры шума и неискаженного изображения неизвестны и не могут быть оценены, часто используется подход, состоящий в аппроксимации выражения (6.31) выражением

$$\hat{F}(u, v) = \left(\frac{1}{H(u, v)} \frac{|H(u, v)|^2}{|H(u, v)|^2 + K} \right) G(u, v), \quad (6.32)$$

где K — определенная константа, прибавляемая ко всем значениям $|H(u, v)|^2$.

Фильтрация по Тихонову

Проблема, заключающаяся в необходимости иметь некоторую информацию относительно искажающей функции, является общей для всех линейных методов восстановления. Применение винеровской фильтрации связано с дополнительной трудностью, состоящей в том, что энергетические спектры неискаженного изображения и шума также должны быть известны.

Использование приближения (6.32) часто позволяет получать отличные результаты. Однако использование константы в качестве оценки для отношения энергетических спектров не всегда приводит к удовлетворительному решению задачи.

Применение метода, рассматриваемого в этой главе, требует только знания среднего значения и дисперсии шума. Это является важным преимуществом метода, поскольку, как показано ранее, обычно можно оценить указанные величины на основе заданного искаженного изображения. Другое отличие состоит в том, что винеровская фильтрация основана на минимизации в смысле некоторого статистического критерия и, следовательно, является оптимальной в некотором среднестатистическом смысле. Метод, рассматриваемый в настоящем раз-

деле, обладает тем замечательным свойством, что позволяет получить оптимальный результат для *каждого* конкретного изображения, к которому он применяется.

Используя определение свертки, можно переписать уравнение (6.26) в матрично-векторном виде следующим образом [98]:

$$\mathbf{g} = \mathbf{H}\mathbf{f} + \boldsymbol{\eta}. \quad (6.33)$$

Пусть, например, изображение $g(x, y)$ имеет размеры $M \times N$. Тогда мы можем сформировать вектор $\mathbf{g}$ таким образом, чтобы первые N его элементов были равны значениям в первой строке изображения $g(x, y)$, следующие N элементов были равны значениям во второй строке и т. д. Полученный вектор будет иметь размер $MN \times 1$. Аналогично формируются векторы $\mathbf{f}$ и $\boldsymbol{\eta}$, которые в результате имеют те же размеры. Далее матрица $\mathbf{H}$ имеет размеры $MN \times MN$. Ее элементы задаются значениями ядра h в свертке.

Естественно предположить, что задача восстановления может быть таким способом сведена к задаче линейной алгебры. К сожалению, дело обстоит не так просто. Предположим для примера, что мы работаем с изображениями средних размеров; пусть для определенности $M = N = 512$. Тогда векторы в формуле (6.33) будут иметь размеры 262144, а матрица $\mathbf{H}$ будет иметь размеры 262144×262144 . Работа с векторами и матрицами подобных размеров представляет собой далеко не простую задачу. Дело дополнительно осложняется тем, что задача определения $\mathbf{f}$ из уравнения (6.33) очень чувствительна к шумам. Это происходит от того, что матрица $\mathbf{H}$ является плохо определенной (т. е. ее определитель близок или даже равен нулю), вследствие чего соответствующая линейная задача является в высшей степени некорректной. Ее решение в обычном смысле может вообще не существовать, и даже если оно существует, то является очень неустойчивым по отношению к шуму $\boldsymbol{\eta}$. Решение подобных задач на практике является делом исключительной трудности.

Тем не менее, формулировка задачи восстановления в матричном виде облегчает построение методов восстановления.

Корни метода восстановления, составляющего предмет настоящей главы, лежат в области матричного анализа. Главной проблемой является чувствительность задачи по отношению к шуму. Один из способов преодоления этой трудности состоит в регуляризации задачи, которая достигается заменой исходной задачи на задачу нахождения

экстремума (минимума) некоторого сглаживающего функционала. Эта идея лежит в основе общего подхода к решению некорректных задач, который был предложен и развит школой А. Н. Тихонова [106]. Такой подход называется *тихоновским методом регуляризации*, и рассматриваемый ниже метод является его весьма частным случаем.

В качестве минимизируемого функционала $C[\mathbf{f}]$ можно использовать квадрат нормы лапласиана

$$C[\mathbf{f}] = \sum_{(x=0)}^{(M-1)} \sum_{(y=0)}^{(N-1)} (\nabla^2 f(x, y))^2 \quad (6.34)$$

с дополнительным ограничением (связью) вида

$$\|\mathbf{g} - \mathbf{H}\hat{\mathbf{f}}\|^2 = \|\boldsymbol{\eta}\|^2, \quad (6.35)$$

где $\|\mathbf{w}\|^2 = \mathbf{w}^T \mathbf{w}$ — квадрат евклидовой нормы вектора $\mathbf{w}$, а $\hat{\mathbf{f}}$ — искомая оценка неискаженного изображения. Оператор Лапласа ∇^2 — это сумма вторых частных производных функции f . Напомним, что если $\mathbf{w}$ — n -компонентный вектор, то

$$\mathbf{w}^T \mathbf{w} = \sum_{k=1}^n w_k^2,$$

где w_k — k -я координата вектора $\mathbf{w}$.

Решение оптимизационной задачи (6.34) с условием (6.35) в частотной области дается выражением

$$\hat{F}(u, v) = \left(\frac{H^*(u, v)}{|H(u, v)|^2 + \gamma |P(u, v)|^2} \right) G(u, v), \quad (6.36)$$

где параметр γ (параметр регуляризации) должен быть выбран таким образом, чтобы выполнялось условие (6.35), а функция $P(u, v) = -(u^2 + v^2)$ есть частотное представление оператора Лапласа [98]. Обратим внимание, что при обращении параметра регуляризации γ в нуль выражение (6.36) сводится к инверсной фильтрации.

Ниже приведена итеративная процедура такого выбора параметра регуляризации γ , чтобы выполнялось условие (6.35).

Определим вектор невязки $\mathbf{r}$ следующим образом:

$$\mathbf{r} = \mathbf{g} - \mathbf{H}\hat{\mathbf{f}}. \quad (6.37)$$

Поскольку решение $\hat{F}(u, v)$ (и соответствующий вектор $\hat{\mathbf{f}}$), определяемое по (6.36), есть функция параметра γ , вектор невязки $\mathbf{r}$ также зависит от этого параметра. Оказывается [107], функционал невязки

$$\varphi(\gamma) = \mathbf{r}^T \mathbf{r} = \|\mathbf{r}\|^2$$

является монотонной функцией параметра γ . Параметр γ необходимо выбрать, чтобы выполнялось условие

$$\|\eta\|^2 - a \leq \|\mathbf{r}\|^2 \leq \|\eta\|^2 + a, \quad (6.38)$$

где коэффициент a задает приемлемую точность выполнения условия связи. В случае $a = 0$ имеет место $\|\mathbf{r}\|^2 = \|\eta\|^2$, и ввиду (6.37) условие (6.35) выполняется точно.

Поскольку функционал невязки $\varphi(\gamma)$ является монотонной функцией параметра регуляризации, нахождение искомого значения γ не представляет трудности. Один из алгоритмов состоит в следующем.

1. Задать начальное значение γ .

2. Вычислить $\|\mathbf{r}\|^2$.

3. Если условие (6.38) выполняется, то цель достигнута. В противном случае увеличить значение γ , если $\|\mathbf{r}\|^2 < \|\eta\|^2 - a$, или уменьшить его, если $\|\mathbf{r}\|^2 > \|\eta\|^2 + a$, и повторить шаг 2, используя новую оценку, пересчитанную по (6.36) с новым значением γ .

Для увеличения скорости сходимости можно использовать более продвинутые методы, например, метод касательных Ньютона.

Для того чтобы использовать данный алгоритм, необходимы величины $\|\mathbf{r}\|^2$ и $\|\eta\|^2$. Для вычисления $\|\mathbf{r}\|^2$ заметим, что из (6.37) следует равенство

$$R(u, v) = G(u, v) - H(u, v)F(u, v),$$

и функция $r(x, y)$ может быть получена вычислением обратного Фурье-преобразования от функции $R(u, v)$. Далее

$$\|\mathbf{r}^2\| = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} r^2(x, y). \quad (6.39)$$

Вычисление $\|\eta\|^2$ приводит к интересному результату. Прежде всего, рассмотрим дисперсию шума полного изображения, которую мы оцениваем известным методом выборочного среднего:

$$\sigma_\eta^2 = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (\eta(x, y) - m_\eta)^2,$$

где

$$m_{\eta} = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \eta(x, y)$$

есть выборочное среднее значение шума. Сравнивая два последних выражения с выражением для $\|\eta\|^2$, которое имеет тот же вид, что и (6.39), мы видим, что

$$\|\eta\|^2 = MN(\sigma_{\eta}^2 + m_{\eta}^2).$$

Это очень важный результат, который показывает, что для реализации оптимального алгоритма восстановления достаточно знать лишь среднее значение и дисперсию шума. Эти величины нетрудно оценить при условии, что значения шума и изображения не коррелированы.

Практическое задание к п. 6.3

Выполните программную реализацию и сравнение оптимальных фильтров Винера и Тихонова на примере смазанного изображения лица с веб-камеры.

6.4. Дескрипторы особенностей

Обнаружения и сопоставление точечных особенностей на изображениях является важной задачей компьютерного зрения и находит применение в таких приложениях, как 3D-реконструкция, задачи слежения, поиск в базах данных изображений и т. д.

Особые точки (в разных источниках — *features/characteristic points/local feature points/interest point/локальные особенности*) — говоря неформально — «хорошо различимые» фрагменты изображения. Это точки (пиксели) с характерной (особой) окрестностью — т. е. отличающиеся своей окрестностью от всех соседних точек. Примерами локальных особенностей могут быть вершины углов, изолированные точечные особенности, контуры и т. п. Особые точки описываются дескрипторами — векторами признаков, вычисляемых на основе интенсивности/градиентов или других характеристик точек окрестности.

Используя особые точки, можно анализировать как изображения целиком, так и объекты на них. Хорошие характерные точки позволяют справиться с изменением масштаба, ракурса и перекрытиями сцены или объекта.

Здесь представлены продвинутые, наиболее популярные дескрипторы особенностей, применяемые в современных системах обработки изображений.

Алгоритм SIFT (Scale Invariant Feature Transform)

Алгоритм *SIFT* (*Scale Invariant Feature Transform*), изначально опубликованный канадским профессором Дэвидом Лоуе (David G. Lowe) в 1999 году, широко используется и является основой для многих впоследствии разработанных алгоритмов. Он является сложным для вычисления, но очень точен по сравнению со многими другими методами и, следовательно, хорошо подходит для задач отслеживания и распознавания [108].

Алгоритм предназначен для выделения особых точек, инвариантных к изменению масштаба, поворотам, изменению яркости и положению камеры. Основным моментом в детектировании особых точек по методу *SIFT* [109] является построение пирамиды гауссианов и разностей гауссианов. Гауссианом называется изображение, размытое гауссовым фильтром. Операцию размытия можно представить в виде свертки изображения с гауссовым ядром:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y),$$

где x, y — координаты пикселя, σ — радиус размытия и одновременно параметр масштаба, $L(x, y, \sigma)$ — гауссиан, $I(x, y)$ — исходное изображение, $G(x, y, \sigma)$ — гауссово ядро:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{\sigma^2}}.$$

Разностью гауссианов называют изображение, полученное в результате попиксельного вычитания гауссиана с радиусом размытия σ из гауссиана с радиусом $k\sigma$:

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma).$$

Функция $L(x, y, \sigma)$ формирует так называемое масштабируемое пространство изображения. Параметр σ является параметром масштаба, при $\sigma = 0$ имеем исходное изображение $L(x, y, 0) = I(x, y)$, при увеличении параметра σ получаются все более и более сглаженные версии исходного изображения.

Для построения пирамиды гауссианов и разностей гауссианов все масштабируемое пространство изображения разделяется по шкале масштаба на участки, которые называются октавами (октава соответствует удвоению параметра σ). При этом изображение при переходе от одной октавы к другой уменьшается по каждой координате в два раза. В каждой октаве вычисления ограничиваются некоторым количеством n гауссианов с определенным шагом по радиусу размытия $k_i\sigma$. Значение k_i выбирается так, чтобы получилось фиксированное число n свернутых изображений на октаву $k_i = 2^{i/n}$. С подходящими шагами достраиваются еще два гауссиана (снизу и сверху октавы), что понадобится при нахождении точки экстремума. Затем вычисляется разность гауссианов от смежных размытых по Гауссу изображений в октаве. Описанный процесс иллюстрируется рис. 6.10, из которого видно, что разностных изображений в октаве на единицу меньше количества гауссианов.

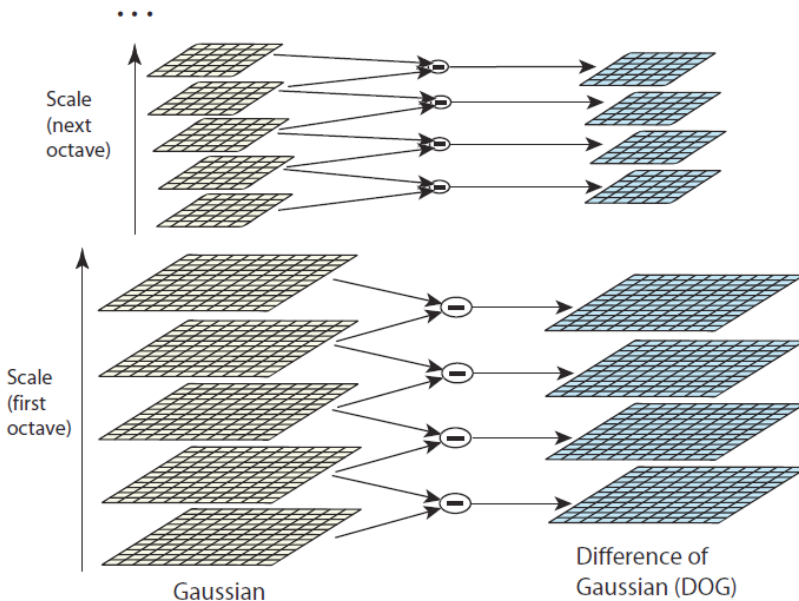


Рис. 6.10. Построение пирамиды разности гауссианов [109]

Как только разность гауссианов изображений получена, ключевые точки определяются как локальные минимумы/максимумы этой разности. Это делается путем сравнения значений разности гауссианов каждого пикселя для его восьми соседей в том же масштабе и девяти соответствующих соседних пикселей в каждом из соседних масштабов (как показано на рис. 6.11). Если значение пикселя является максимумом или минимумом среди всех сравниваемых точек, оно выбирается как кандидат особой точки. Использование масштабной пирамиды при поиске особых точек обеспечивает свойство инвариантности метода *SIFT*.

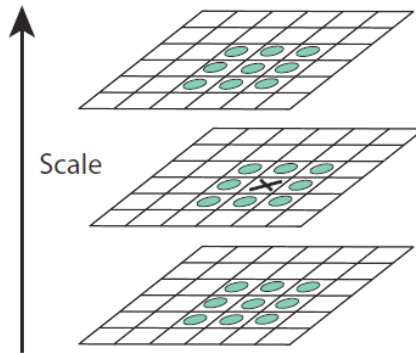


Рис. 6.11. Определение особой точки по локальному экстремуму [109]

При этом локальный экстремум еще нельзя считать особой точкой, необходимо дополнительно провести несколько проверок пригодности для того, чтобы, во-первых, удалить путем пороговой обработки малоcontrastные точки, во-вторых, на основе анализа матрицы Гессе функции $L(x, y, \sigma)$ оставить только существенные особенности изображения и заодно обеспечить инвариантность особых точек к аффинным преобразованиям изображения. Полученное таким образом множество особых точек идет на дальнейшую обработку.

Следующий шаг алгоритма *SIFT* состоит в вычислении ориентаций особых точек. Ориентация особой точки будет определяться путем вычисления градиентов в ее соседних точках. Все вычисления будут производиться в пирамиде гауссиан, в масштабе, наиболее близком к масштабу особой точки. Величина градиента m и направление градиента θ в точке с координатами (x, y, σ) определяются следующим образом:

$$m(x, y, \sigma) = \sqrt{(L(x+1, y, \sigma) - L(x-1, y, \sigma))^2 + (L(x, y+1, \sigma) - L(x, y-1, \sigma))^2},$$

$$\theta(x, y, \sigma) = \tan^{-1} \left(\frac{L(x, y+1, \sigma) - L(x, y-1, \sigma)}{L(x+1, y, \sigma) - L(x-1, y, \sigma)} \right).$$

За окрестность особой точки, в которой будут рассматриваться градиенты, берется окрестность радиуса по известному правилу «трех сигм».

Направление особой точки определяется из гистограммы направлений θ , состоящей из 36 компонентов, которые равномерно покрывают промежуток в 360° по 10° соответственно. Каждый градиент вносит вклад, равный $m(x, y, \sigma) * G(x, y, \sigma)$, в ту компоненту гистограммы, которая входит в промежуток, содержащий направление $\theta(x, y, \sigma)$.

Теперь рассмотрим определение дескрипторов в методе *SIFT*. Дескриптор *SIFT* — это вектор, компонентами которого являются значения гистограммы ориентаций градиентов (*histogram of oriented gradients* — *HOG*) в заданной окрестности особой точки. Как и направление точки, дескриптор определяется на ближайшем по масштабу гауссиане. В первую очередь окно для расчетов *HOG* поворачивается на угол направления особой точки, чем достигается инвариантность дескрипторов относительно поворота. Вычисление дескрипторов ведется в той же окрестности, что и ориентация особой точки. Прямоугольная окрестность делится на 4×4 регионов и для каждого региона вычисляются гистограммы на восьми направлениях. В результате каждый *SIFT*-дескриптор особой точки содержит $4 \times 4 \times 8$ элементов. На рис. 6.12 для простоты описанный процесс иллюстрируется на примере дескриптора меньшей размерности. В заключение полученный дескриптор нормализуется для обеспечения устойчивости к изменениям яркости изображения.

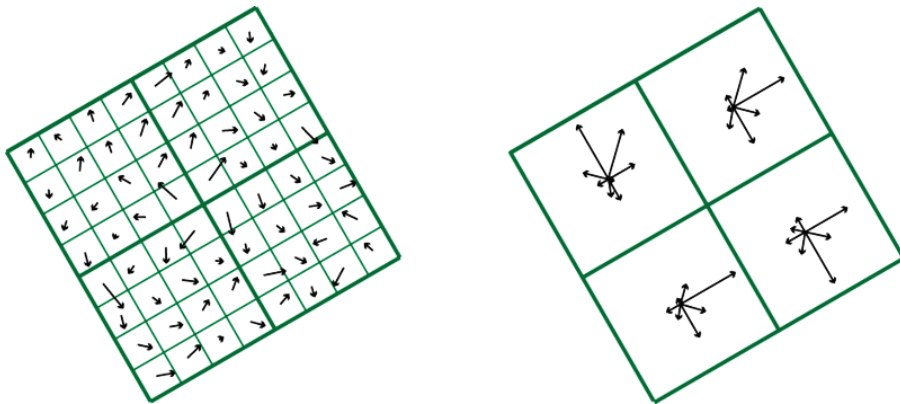


Рис. 6.12. Дескриптор особой точки размерности $2 \times 2 \times 8$. Градиенты, направление их пикселей (слева) и сформированный дескриптор (справа) [110]

Алгоритм SURF (Speeded-UP Robust Features)

Алгоритм *SURF* был изначально предложен в 2006 году и во многом является продолжением алгоритма *SIFT*. Создатели *SURF* были заинтересованы в том, чтобы различные компоненты алгоритма *SIFT* могли быть заменены на более эффективные в плане вычисления, которые могли бы обеспечить аналогичную или повышенную производительность в первую очередь в задачах распознавания. Получившиеся функции не только способны на более быстрое вычисление, но и во многих случаях их немного более простая природа приводит к большей устойчивости к изменениям ориентации или освещения, чем в *SIFT*.

Если в алгоритме *SIFT* используются разности гауссиан, чтобы построить пирамиду разностей и впоследствии найти локальный экстремум, то в методе *SURF* все основывается на вычислении детерминанта матрицы Гессе (гессиана).

Матрица Гессе выглядит следующим образом:

$$H(f(x, y)) = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix}.$$

Для матрицы H вычисляется детерминант:

$$\det(H) = \frac{\partial^2 f}{\partial x^2} \frac{\partial^2 f}{\partial y^2} - \left(\frac{\partial^2 f}{\partial x \partial y} \right)^2,$$

где H — матрица Гессе, $f(x, y)$ — сглаженная функция яркости.

В методе *SURF* вычисление матрицы Гессе сводится к вычислению лапласиана от гауссиана. Элементы матрицы вычисляются как сумма произведений пикселей изображения на определенные фильтры (рис. 6.13).

Однако [111] этот фильтр имеет большой разброс детерминанта при вращении изображения и особенно «проседает» в районе поворота в 45° . И еще данный фильтр имеет непрерывный характер и почти все пиксели фильтра имеют различную друг от друга яркость, что не дает использовать единую матрицу. По этой причине на практике используют бинаризованный метод аппроксимации лапласиана гауссиан, показанный на рис. 6.14.

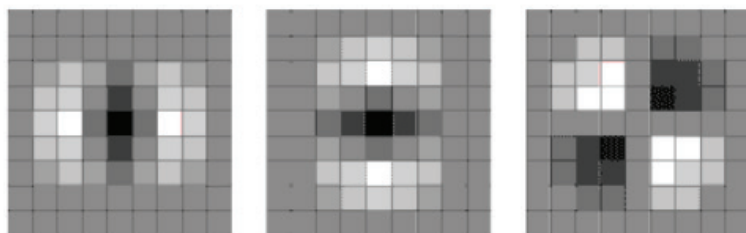


Рис 6.13. Дискретизированные фильтры для нахождения элементов матрицы Гессе [111]

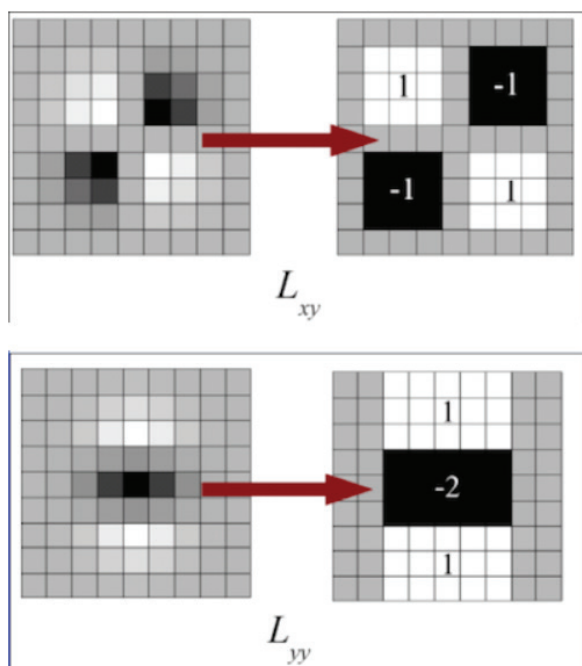


Рис. 6.14. Фильтры, используемые для нахождения матрицы Гессе в *SURF* [112]

Блочные фильтры размера 9×9 на рисунках выше являются приближением гауссиана с $\sigma = 1, 2$ и представляют наименьший размер (т. е. самое большое разрешение) для вычисления карт отклика большого двоичного объекта.

Как уже говорилось ранее, использование гессииана обеспечивает инвариантность по отношению к повороту, но не к масштабу. По этой причине производится фильтрация изображения по гауссовому ядру, в результате чего матрица Гессе $H(x, y, \sigma)$ определяется как:

$$H(x, y, \sigma) = \begin{bmatrix} L_{xx}(x, y, \sigma) & L_{xy}(x, y, \sigma) \\ L_{xy}(x, y, \sigma) & L_{yy}(x, y, \sigma) \end{bmatrix},$$

где $L_{xx}(x, y, \sigma)$, $L_{xy}(x, y, \sigma)$, $L_{yy}(x, y, \sigma)$ — свертки аппроксимации второй производной Гауссова ядра с изображением I .

Таким образом, в *SURF*, гессиан вычисляется так:

$$\det(H_{approx}) = D_{xx} D_{yy} - (0,9 D_{xy})^2,$$

где D_{xx} , D_{yy} , D_{xy} — свертки по фильтрам, изображенным на рис. 6.14. Коэффициент 0,9 имеет теоретическое обоснование [111] и используется для того, чтобы корректировать характер вычислений.

Таким образом, метод для нахождения особых точек проверяет все пиксели изображения и ищет максимум гессиана. В методе задается пороговое значение гессиана и, если вычисленное значение будет больше порогового, то пиксель будет рассматриваться как кандидат на становление особой точкой. Так же стоит отметить, что метод способен распознавать как темные пятна на светлом фоне, так и светлые пятна на темном фоне.

Следующим этапом будет нахождение локального экстремума гессиана. Алгоритм аналогичен тому, что был при нахождении особой точки в методе *SIFT*, и пиксель сравнивается с восемью своими соседями и девятью гессианами на уровень выше и ниже. Из чего становится понятно, что для определения локального экстремума необходимо, чтобы октава содержала не менее трех фильтров, иначе нахождение экстремума становится невозможным.

Для достижения инвариантности в *SIFT* и нахождения ориентации особой точки были использованы гистограммы ориентаций градиентов, метод *SURF* в свою очередь использует фильтр Хаара (рис. 6.15).



Рис 6.15. Фильтры Хаара. Черные области имеют значения -1 , белые $+1$ [111]

Сначала рассчитываются значения перепада яркости в направлении x и y в круговой окрестности вокруг особой точки с радиусом bs ,

где s — масштаб ключевой точки. Вычисляется сумма вертикальных и горизонтальных значений в области сканирования, а затем меняется ориентация области (добавляем $\pi/3$) и пересчитывается, пока не найдется ориентация с наибольшим значением суммы, эта ориентация и будет являться основной для дескриптора (рис. 6.16).

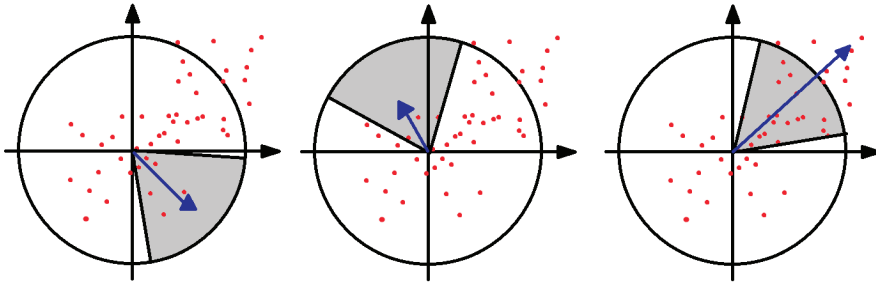


Рис. 6.16. Вычисление направления дескриптора [112]

Теперь рассмотрим построение дескриптора метода *SURF*. Сначала строится квадратная область, центрированная вокруг особой точки и ориентированная по значению большей суммы, размер этого окна будет составлять $20s$. Затем эта область делится на 16 квадратных регионов, после чего для каждого региона берется регулярная сетка 5×5 и для каждой точки этой сетки вычисляется значение градиента при помощи фильтра Хаара. Размер фильтра Хаара берется равным $2s$ (рис. 6.17).

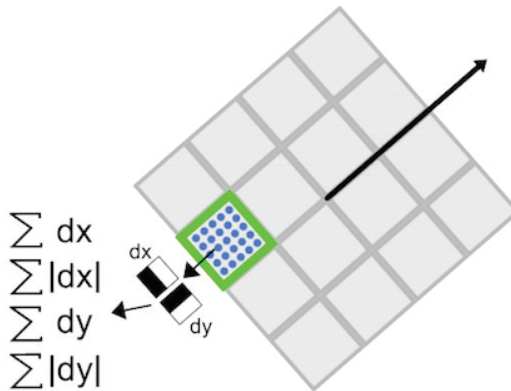


Рис. 6.17. Определение дескриптора изображения методом *SURF* [111]

Затем в пределах каждого региона вычисляется сумма откликов фильтров Хаара в направлении x и y , кроме того, вычисляются сум-

мы их абсолютных значений. В итоге для каждого региона получается вектор:

$$v = (\sum dx, \sum dy, \sum |dx|, \sum |dy|)^T.$$

Следовательно, каждый регион имеет четырехмерный вектор дескриптора v . Объединяя это для всех 4×4 регионов, получим в итоге 64-мерный дескриптор одной особой точки. Для сравнения: *SIFT* имеет 128-мерный вектор, и это одна из причин, почему *SURF* оказывается более быстрым алгоритмом, чем *SIFT*.

Алгоритм ORB (Oriented FAST and Rotated BRIEF)

Метод ORB (*Oriented FAST and Rotated BRIEF*) был представлен в 2011 году, и в его основе лежит комбинация таких алгоритмов, как детектор *FAST* (Feature from Accelerated Segment Test) и дескриптор *BRIEF* (Binary Robust Independent Elementary Features).

Для многих приложений очень важна скорость вычисления. Это особенно верно для задач, которые, как ожидается, будут выполняться в режиме реального времени на видеоданных, таких как приложения дополненной реальности или робототехники. По этой причине метод *SURF* был разработан с целью обеспечения возможностей, аналогичных *SIFT*, но на гораздо более высокой скорости. Точно так же функция *ORB* была создана с целью предоставления еще более высокоскоростной альтернативы *SIFT* и *SURF*. Алгоритм был представлен в 2011 году и в качестве детектора он использует алгоритм *FAST* (Feature from Accelerated Segment Test) для обнаружения набора особых точек. Быстрые методы несмотря на скорость имеют ряд недостатков. Одним из таких недостатков является то, что *FAST* имеет тенденцию реагировать на углы. Для того чтобы избежать этого в данном методе также используется фильтр Харриса, где рассматриваются производные яркости изображения для исследования изменений яркости по множеству направлений [113].

Алгоритм *FAST* предназначен для обнаружения особых точек, и, по сравнению с остальными рассмотренными алгоритмами, является только детектором. Первоначально он был предложен Ростеном и Драммондом и основан на идее прямого сравнения яркости между точкой P и набором точек на маленьком круге вокруг нее.

Первая особенность метода *FAST* при анализе соседей P состоит в том, что он использует точки только на кольце вокруг P . Вторая особенность в том, что отдельные точки на кольце классифицируются как темнее P , светлее P или аналогичные P . Эта классификация рассматривается с порогом t , так что если находится N точек, интенсивность которых меньше, чем $I_p - t$ или больше чем $I_p + t$, то она может считаться особой. Изначально рассматривается окружность из 16 пикселей вокруг точки кандидата P . Далее нужно сравнить интенсивность точек по вертикали и горизонтали. Примеры этих точек находятся на рис. 6.18 под номерами 0, 4, 8, 12. И только в случае, когда три из этих точек будут либо темнее, либо светлее P , проверяются все 16 точек. Данная процедура производится с целью отсеять как можно больше кандидатов и быстрее обработать изображение. Согласно экспериментам [114], наименьшее значение N , при котором особые точки стабильно обнаруживаются, равно 9 (рис. 6.18).

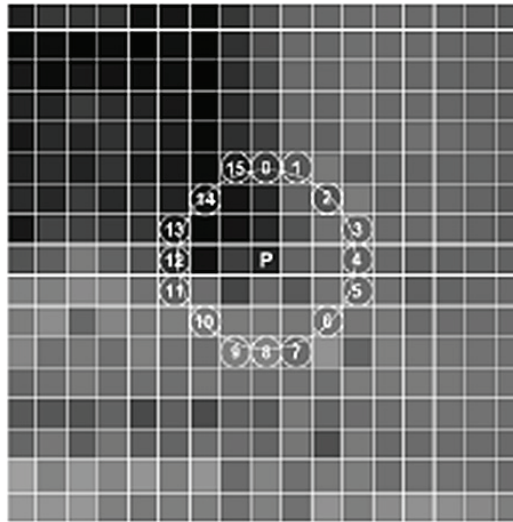


Рис. 6.18. Проверка точки P алгоритмом *FAST* [4, с. 538]

Следующим шагом алгоритма *ORB* будет определение угла ориентации особой точки. Для этого сначала вычисляются моменты яркости окрестности особой точки:

$$m_{pq} = \sum_{x,y} x^p y^q I(x,y),$$

где x, y — пиксельные координаты, $I(x, y)$ — яркость, после чего вычисляется сам угол особой точки:

$$\theta = \text{atan}^2(m_{01}, m_{10}).$$

После получения угла ориентации особой точки можно переходить к вычислению ее характеристик. В качестве дескриптора метод использует *BRIEF*, основная идея которого состоит в том, что элемент описывается как серия тестов, каждый из которых просто сравнивает один пиксель в области элемента с некоторым другим, давая простой двоичный результат (то есть 0 или 1). Дескриптор *BRIEF* является просто результатом n таких тестов, сгруппированных в битовую строку.

Чтобы дескриптор не был слишком чувствителен к шуму, исходное изображение предварительно сглаживается путем свертки с ядром Гаусса. Поскольку дескрипторы являются двоичными строками, они не только могут быть быстро вычислены и эффективно сохранены, но также могут быть чрезвычайно эффективно сопоставлены друг с другом.

Существует много способов для генерации фактических пар, которые будут сопоставлены для формирования дескриптора *BRIEF*. Один из лучших способов — просто случайным образом сгенерировать все пары следующим способом (см. рис. 6.19): сначала получить точку из гауссовского распределения с центром в точке кандидата, а затем получить вторую точку из гауссовского распределения с центром в первой. Бинарный тест между точками X и Y в окрестности тестируемой точки P имеет следующий вид:

$$\tau(P, X, Y) = \begin{cases} 1, & I(X) < I(Y), \\ 0, & \text{иначе.} \end{cases}$$

Алгоритм AKAZE (Accelerated-KAZE)

Алгоритм *KAZE* и его ускоренная модификация *A-KAZE* 2013 года является новым методом 2D-обнаружения и описания признаков, который работает лучше, чем *SIFT* и *SURF*. Метод получил широкую популярность вследствие того, что распространяется свободно и имеет открытые исходные коды. Алгоритм *KAZE* создали Пабло Ф. Алькантарилья, Эдриен Бартоли и Эндрю Дж. Дэвисон [115].

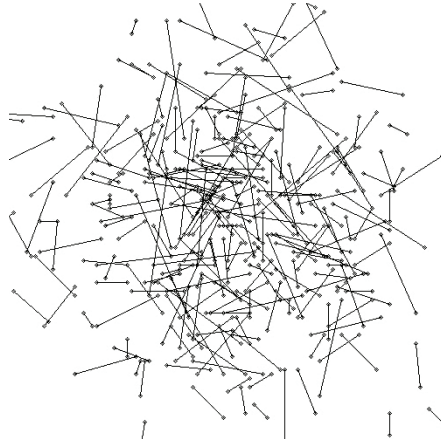


Рис. 6.19. Визуализация тестов от пикселя к пикселю, которые в совокупности составляют один дескриптор *BRIEF*, каждая линия соединяет пару тестируемых пикселей [108, с. 555]

Алгоритм *AKAZE* основан на нелинейной диффузионной фильтрации, но его нелинейные многомасштабные пирамиды построены с использованием вычислительно эффективной среды под названием *Fast Explicit Diffusion* [116]. Детектор основан на детерминанте матрицы Гессе и применении нелинейного коэффициента масштабирования, который позволяет увеличить скорость нахождения особой точки по сравнению с Гауссовой пирамидой. Данный коэффициент вычисляется по изменению яркости на изображении при его масштабировании.

Для каждой октавы L^i в пирамиде вычисляется определитель Гессеана.

$$L_{Hessian}^i = \sigma_{i,norm}^2 (L_{xx}^i L_{yy}^i - L_{xy}^i L_{yx}^i).$$

Здесь $\sigma_{i,norm}^2 = \frac{\sigma_i}{2^{s^i}}$ — нормализованный относительно масштаба коэффициент для вычисления определителя Гессеана с учетом размера октавы σ_i .

Инвариантность ориентации достигается с помощью фильтра Шарра [117], за счет которого вычисляются значения производных второго порядка с шагом $\sigma_{i,norm}$. Этот фильтр позволяет учитывать ориентацию особой точки. При помощи него ищем в октаве все такие точки, чтобы значение фильтра в них оказалось выше заданного порога и наибольшим по сравнению с остальными в окрестности точ-

ки 3×3 пикселей. Максимумы откликов детектора при сравнении с соседними октавами $i+1$ и $i-1$ в окне размером $\sigma_i \times \sigma_i$ выбираются как особые точки.

Дескриптор основан на алгоритме *MLDB* (*Modified Local Difference Binary*), который также является высокоэффективным. Изначально использовавшийся дескриптор *LDB* основывался на рассмотренном выше методе *BRIEF*, но у метода были проблемы с инвариантностью. Для решения этих проблем в *AKAZE* используется более совершенная его версия *MLDB*, в которой к сравнениям яркостных показателей метода были добавлены сравнения градиентов яркости по осям x и y . Элементы метода инварианты к масштабу, вращению, аффинным преобразованиям и обладают большой различимостью в различных масштабах из-за нелинейных масштабных пространств.

В отличие от *LDB* в *MLDB* тесты проводятся не между средними значениями показателей всех пикселей в области, а между определенным заданным их количеством в зависимости от размера. Что также ускоряет работу дескриптора.

В итоге был разработан быстрый метод, в котором найденные особые точки и их дескрипторы удовлетворяют высоким показателям точности при сравнении изображений.

Практическое задание к п. 6.4

Выполните программную реализацию дескрипторов особых точек с использованием программных средств библиотеки OpenCV [108] для изображения лица с веб-камеры.

6.5. Обработка видеопоследовательности

В медицине получило широкое распространение применение методов цифровой обработки видеоизображений. Актуальность такого направления имеет два аспекта. Во-первых, это наличие качественной и недорогой цифровой видеотехники. Во-вторых, быстрое развитие компьютерной техники и математических методов анализа изображе-

ний и компьютерного зрения. Обработка видеоизображений в задачах биомедицины при помощи современных методов компьютерного зрения может применяться для:

- автоматического обнаружения событий;
- обнаружения объектов и измерения их параметров;
- поиска объектов по образцу;
- выявления и исследования характеристик объектов по видеоизображению;
- анализа эмоций, определения психоэмоционального состояния (*affective computing*).

Оптический поток

В биомедицинских системах компьютерного зрения и обработки изображений иногда возникает задача определения перемещений объектов в трехмерном пространстве с помощью видеокамеры. При этом по последовательности кадров необходимо измерить пространственные перемещения, которые происходят с объектом с течением времени. На практике зачастую достаточно найти смещения двухмерных проекций объектов в плоскости кадра.

Оптический поток (*optical flow*) позволяет определить смещение объекта по отношению к его же положению на предыдущем кадре за то время, которое прошло между фиксацией кадров. Для нахождения оптического потока можно воспользоваться готовой протестированной и оптимизированной реализацией одного из алгоритмов из библиотеки *OpenCV* [108]. При этом, однако, очень важно и полезно разобраться в математической сущности метода.

В основе метода оптического потока лежит вычисление частных производных по горизонтальному и вертикальному направлениям изображения. Как можно увидеть далее, одних только производных недостаточно, чтобы определить смещения. Именно поэтому на базе одной очевидной идеи появилось множество методов. Рассмотрим наиболее простой и, тем не менее, работоспособный метод Лукаса-Канаде (*Lucas-Kanade*), предложенном в 1981 году Брюсом Лукасом и Такео Канаде [118].

Метод Лукаса-Канаде

При выводе соотношений метода Лукаса-Канаде предполагается, что пиксели, относящиеся к одному и тому же объекту, могут сместиться в какую-либо сторону, но их значение (яркости) остаются неизменными. Конечно же, это предположение весьма приближенно соответствует реальности, потому что от кадра к кадру могут меняться условия освещения и яркость самого движущегося объекта. С этим допущением связаны некоторые проблемы, но при этом оно достаточно хорошо работает на практике.

На математическом языке это допущение выражается следующим образом:

$$I(x, y, t) = I(x + u_x, y + u_y, t + 1). \quad (6.40)$$

Здесь I — это яркость пикселя (x, y) в момент времени t . Более подробно x и y — это координаты пикселя в плоскости кадра, u_x и u_y — это смещение пикселя, а t — это номер кадра в последовательности. Для простоты изложения будем считать, что между двумя соседними кадрами интервал по времени равен единице.

Запишем разложение в ряд Тейлора для функции $I(x + u_x, y + u_y, t)$, отбросим все старшие производные и получим следующее соотношение:

$$I(x + u_x, y + u_y, t) = I(x, y, t) + \frac{\partial I(x, y, t)}{\partial x} u_x + \frac{\partial I(x, y, t)}{\partial y} u_y.$$

Условие (6.40) эквивалентно равенству $I(x + u_x, y + u_y, t) = I(x, y, t + 1)$, подстановка в которое вышеприведенного разложения приводит к уравнению:

$$I(x, y, t) - I(x, y, t + 1) + \frac{\partial I(x, y, t)}{\partial x} u_x + \frac{\partial I(x, y, t)}{\partial y} u_y = 0.$$

Поскольку между двумя кадрами интервал времени единичный, то разность $I(x, y, t) - I(x, y, t + 1)$ есть не что иное, как приближение производной по времени:

$$\frac{\partial I(x, y, t)}{\partial t} + \frac{\partial I(x, y, t)}{\partial x} u_x + \frac{\partial I(x, y, t)}{\partial y} u_y = 0. \quad (6.41)$$

Таким образом, получено одно уравнение относительно двух неизвестных смещений u_x и u_y . С этим фактом связано разнообразие подходов к решению задачи.

Согласно методу Лукаса-Канаде предположим, что *соседние пиксели смещаются на одинаковое расстояние*. Возьмем окрестность точки (x, y) , например, размером 5×5 пикселей, и условимся, что для каждого из 25 пикселей u_x и u_y равны. Тогда вместо одного уравнения получим линейную систему из 25 уравнений с двумя неизвестными и будем решать ее методом наименьших квадратов, т. е. будем искать такие u_x и u_y , которые минимизируют ошибку:

$$E(u_x, u_y) = \sum_{i,j} g(x_i, y_i) \left[\frac{\partial I(x, y, t)}{\partial t} + \frac{\partial I(x, y, t)}{\partial x} u_x + \frac{\partial I(x, y, t)}{\partial y} u_y \right]^2 \rightarrow \min.$$

Здесь g — это функция, определяющая весовые коэффициенты для пикселей окрестности, а суммирование производится по всей окрестности. Самый распространенный вариант — это взять в качестве g двумерную функцию Гаусса (6.9), которая дает наибольший вес центральному пикселю и все меньший по мере удаления от центра.

В данном случае система нормальных уравнений метода наименьших квадратов в матричной форме имеет следующий вид.

$$A\vec{u} = \vec{b},$$

$$\text{где } \vec{u} = \begin{bmatrix} u_x \\ u_y \end{bmatrix},$$

$$A = \begin{bmatrix} \sum_{i,j} g(x_i, y_i) \left(\frac{\partial I}{\partial x} \right)^2 & \sum_{i,j} g(x_i, y_i) \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} \\ \sum_{i,j} g(x_i, y_i) \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} & \sum_{i,j} g(x_i, y_i) \left(\frac{\partial I}{\partial y} \right)^2 \end{bmatrix}, \vec{b} = - \begin{bmatrix} \sum_{i,j} g(x_i, y_i) \frac{\partial I}{\partial t} \frac{\partial I}{\partial x} \\ \sum_{i,j} g(x_i, y_i) \frac{\partial I}{\partial t} \frac{\partial I}{\partial y} \end{bmatrix}.$$

Если матрица A обратима (имеет ранг 2), можем вычислить смещения u_x и u_y , которые минимизируют ошибку E :

$$\vec{u} = A^{-1} \vec{b}.$$

Поскольку в нахождении смещения каждого пикселя участвуют пиксели из окрестности, при реализации этого метода для оптимизации вычислений необходимо предварительно вычислить производные для всех пикселей по горизонтали и вертикали.

К недостаткам описанного метода определения оптического потока относятся:

- 1) принятие только первого линейного приближения при разложении функции яркости в ряд Тейлора;
- 2) метод по своей природе является локальным, поэтому невозможно определить смещения внутри достаточно больших (больше размера локальной окрестности) равномерно окрашенных участков кадра;
- 3) некоторые текстуры на изображении дают вырожденную матрицу A , для которой не может быть найдена обратная матрица, поэтому для таких текстур метод не позволяет определить смещение.

Первая проблема решается путем итеративного уточнения вычислений смещений. Идея такого уточнения для этой пары кадров (назовем их F_i и F_{i+1}) состоит в следующем. Вычислив смещения (u_x^1, u_y^1) на первой итерации, мы перемещаем каждый пиксель кадра F_{i+1} в противоположную сторону $(-u_x^1, -u_y^1)$ так, чтобы это смещение компенсировать. На следующей итерации вместо исходного кадра F_{i+1} мы будем использовать его искаженный вариант F_{i+1}^1 . И так далее, пока на очередной k -ой итерации все полученные смещения (u_x^k, u_y^k) не окажутся меньше заданного порогового значения. Итоговое смещение для каждого конкретного пикселя получаем как сумму его смещений на всех итерациях.

Относительно второй и третьей проблем оказывается, что на реальных кадрах обширные однородные участки и вырожденные (экзотические) текстуры встречаются нечасто, хотя эти особенности все же вносят дополнительное погрешности в формирование оптического потока.

Здесь описаны основы метода Лукаса-Канаде — одного из простейших дифференциальных методов нахождения оптического потока. Существует множество других интересных методов, которые дают более надежные результаты. Хотя проблема нахождения оптического потока изучается уже несколько десятилетий, методы все еще продолжают совершенствоваться. Работа продолжается в виду того, что при более близком рассмотрении проблема оказывается весьма непростой, а от качества определения смещений в обработке видео и изображений зависит устойчивость и эффективность многих других алгоритмов.

Метод главных компонент

Предположим, что для извлечения полезной информации из полученной видеопоследовательности, состоящей из N кадров, для каждого кадра формируется вектор размерности n : $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$, где T означает операцию транспонирования. Например, в методе дистанционной фотоплетизмографии для формирования такого вектора в каждом RGB -кадре видеопоследовательности используются средние значения интенсивности зеленого канала в n прямоугольных фрагментах изображения поверхности кожи пациента.

Далее при использовании матричных обозначений считается, что все векторы являются вектор-столбцами (т. е. матрицами порядка $n \times 1$). Предположим, что эти векторы являются реализациями случайной величины и можно говорить о *векторе математического ожидания* и *ковариационной матрице* случайного вектора $\mathbf{x}$.

Вектор математического ожидания для генеральной совокупности определяется как

$$\mathbf{m}_x = E\{\mathbf{x}\}, \quad (6.42)$$

где $E\{\cdot\}$ есть ожидаемое значение аргумента, а индекс означает, что $\mathbf{m}$ связан с генеральной совокупностью векторов $\mathbf{x}$. Напомним, что ожидаемое значение вектора или матрицы формируется как набор независимых математических ожиданий их компонентов.

Ковариационная матрица для генеральной совокупности векторов определяется как

$$\mathbf{P}_x = E\{(\mathbf{x} - \mathbf{m}_x)(\mathbf{x} - \mathbf{m}_x)^T\}.$$

Поскольку $\mathbf{x}$ есть n -мерный вектор, то $(\mathbf{x} - \mathbf{m}_x)(\mathbf{x} - \mathbf{m}_x)^T$ и $\mathbf{P}_x$ — матрицы порядка $n \times n$. Элемент p_{ii} матрицы $\mathbf{P}_x$ есть дисперсия x_i , т. е. i -й компоненты векторов $\mathbf{x}$ генеральной совокупности, а элемент p_{ij} матрицы $\mathbf{P}_x$ есть ковариация компонент x_i и x_j этих векторов. Матрица $\mathbf{P}_x$ является действительной и симметричной относительно главной диагонали. Если компоненты x_i и x_j являются некоррелированными, то значение их ковариации равно нулю и, следовательно, $p_{ij} = p_{ji} = 0$. Все эти определения при $n = 1$ сводятся к известным одномерным эквивалентам. Напомним, что дисперсия скалярной случайной величины x , имеющей среднее значение m , определяется как $E\{(x - m)^2\}$. Ковариация (второй смешанный момент) двух случайных величин x_i и x_j определяется как $E\{(x_i - m_i)(x_j - m_j)\}$.

На основании выборки K векторов из генеральной совокупности приближенная оценка вектора математического ожидания находится с помощью обычного усреднения:

$$\mathbf{m}_x = \frac{1}{K} \sum_{k=1}^K \mathbf{x}_k. \quad (6.43)$$

Аналогично, раскрывая произведение $(\mathbf{x} - \mathbf{m}_x)(\mathbf{x} - \mathbf{m}_x)^T$ и используя равенства (6.42) и (6.43), получаем следующую выборочную оценку для ковариационной матрицы:

$$P_x = \frac{1}{K} \sum_{k=1}^K \mathbf{x}_k \mathbf{x}_k^T - \mathbf{m}_x \mathbf{m}_x^T.$$

Поскольку матрица P_x является действительной и симметричной, то для нее всегда существует ортонормированный базис, состоящий из n собственных векторов [119]. Пусть $\mathbf{e}_i$ и λ_i , $i = 1, 2, \dots, n$, — набор собственных векторов и соответствующие им собственные значения матрицы P_x , которые для удобства упорядочим по убыванию, так что $\lambda_j \geq \lambda_{j+1}$ для $j = 1, 2, \dots, n-1$. Пусть матрица A состоит из собственных векторов P_x , расположенных по строкам таким образом, что в первой строке записывается вектор, которому отвечает наибольшее собственное значение, а в последней — собственный вектор, соответствующий наименьшему собственному значению. Матрица A будет использоваться в качестве матрицы преобразования, которое отображает векторы $\mathbf{x}$ в векторы $\mathbf{y}$ по следующему закону:

$$\mathbf{y} = A(\mathbf{x} - \mathbf{m}_x). \quad (6.44)$$

Это выражение называется *преобразованием Карунена — Лоэва*, которое обладает рядом интересных и полезных свойств, описанных ниже.

Нетрудно показать, что получаемые в результате такого преобразования векторы $\mathbf{y}$ имеют нулевое математическое ожидание, т. е.

$$\mathbf{m}_y = E\{\mathbf{y}\} = \mathbf{0}.$$

Из элементарной теории матриц следует, что ковариационная матрица генеральной совокупности векторов $\mathbf{y}$ выражается через матрицы A и P_x следующим образом:

$$P_y = A P_x A^T.$$

Более того, учитывая способ построения матрицы A , ковариационная матрица P_y является диагональной, и элементы, находящиеся на главной диагонали, есть собственные значения матрицы P_x , т. е.

$$P_y = \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n \end{bmatrix}$$

Все элементы этой ковариационной матрицы, расположенные вне главной диагонали, равны 0, поэтому компоненты векторов y являются некоррелированными. Учитывая, что все λ_j являются собственными значениями матрицы P_x , а элементы любой диагональной матрицы, находящиеся на ее главной диагонали, — собственными значениями этой матрицы [119], делаем вывод, что матрицы P_x и P_y имеют совпадающий набор собственных значений.

Другое важное свойство преобразования *Карунена — Лозва* связано с восстановлением x по y . Поскольку строки матрицы A представляют собой ортонормированные векторы, отсюда следует, что $A^{-1} = A^T$, и значит, любой вектор x может быть восстановлен по соответствующему вектору y с помощью соотношения

$$x = A^T y + m_x. \quad (6.45)$$

Предположим, однако, что вместо использования всех собственных векторов P_x строится матрица преобразования A_k , состоящая лишь из k собственных векторов, которые отвечают k наибольшим собственным значениям; т. е. матрица A_k имеет размеры $k \times n$. Тогда векторы y будут иметь размерность k , и восстановление по формуле (6.45) перестанет быть точным. С использованием матрицы A_k восстанавливаться будет вектор

$$\hat{x} = A_k^T y + m_x \quad (6.46)$$

Можно показать, что средний квадрат ошибки e_{ms} между x и $\hat{x}$ задается выражением

$$e_{ms} = \sum_{j=1}^n \lambda_j - \sum_{j=1}^k \lambda_j = \sum_{j=k+1}^n \lambda_j. \quad (6.47)$$

Первое равенство в (6.47) указывает, что ошибка равна 0, если $k = n$ (т. е. когда в преобразовании используются все собственные векторы ковариационной матрицы). Коль скоро значения λ_j в этом построении

монотонно убывают (при увеличении j), из соотношения (6.47) также видно, что при заданном k ошибку можно минимизировать, выбирая k собственных векторов, которым соответствуют наибольшие собственные значения. Таким образом, преобразование *Карунена — Лоэва* оптимально в том смысле, что оно минимизирует средний квадрат ошибки между векторами $\mathbf{x}$ и их приближениями $\hat{\mathbf{x}}$. Учитывая лежащий в основе преобразования *Карунена — Лоэва* принцип использования собственных векторов, отвечающих наибольшим собственным значениям, его также называют приведением к *главным компонентам*.

Важным применением метода главных компонент является задача фильтрации шума и усиление полезного сигнала. Идея такого применения заключается в том, что при преобразовании *Карунена — Лоэва*, как правило, компоненты полезного сигнала оказываются сосредоточенными в подпространстве векторов с наибольшими собственными значениями, а компоненты шума описываются векторами с малыми собственными значениями. Поэтому для удаления шума достаточно отбросить в векторах $\mathbf{y}$, полученных по (6.44), компоненты, отвечающие малым собственным значениям и применить обратное преобразование (6.46). Такое свойство метода главных компонент используется при решении задачи усиления полезного сигнала в методе дистанционной фотоплетизмографии (см. ниже).

Метод главных компонент применяется также для достижения инвариантности описания объекта на изображении безотносительно к сдвигу, масштабированию и повороту, поскольку представление и описание объектов должны быть максимально независимы от этих изменений. Пусть имеется некоторый объект на изображении, и предположим, что его размеры, местоположение и ориентация могут произвольно меняться. Каждую точку области объекта (или ее границы) можно рассматривать как двухкомпонентный вектор $\mathbf{x} = (x_1, x_2)^T$, где x_1 и x_2 — значения координат этой точки по осям x_1 и x_2 . Совокупность всех точек области или границы образует выборку двумерных векторов, которая используется для вычисления вектора математического ожидания $\mathbf{m}_x$ и ковариационной матрицы $\mathbf{P}_x$, как это делалось выше. Один собственный вектор $\mathbf{P}_x$ совпадает с направлением максимальной дисперсии (разброса данных) выборки, а другой будет ему перпендикулярен, поскольку собственные вектора всегда ортогональны друг другу. Применительно к текущему обсуждению преобразование к главным компонентам согласно (6.44) делает две вещи:

- 1) переносит начало координат в центр тяжести (центроид) выборки, поскольку из каждого $\mathbf{x}$ вычитается $\mathbf{m}_x$;
- 2) строит векторы $\mathbf{y}$ путем поворота исходных векторов $\mathbf{x}$, раскладывая данные по направлениям собственных векторов.

Определяя новую систему координат (y_1, y_2) так, что направление оси y_1 совпадает с первым собственным вектором, а оси y_2 — со вторым, получим изображение, в котором преобладающие направления данных совпадают с координатными осями. Независимо от сдвига и поворота объекта будет получен один и тот же результат при условии, что все точки области или границы претерпевают одинаковые изменения. Для нормализации результата относительно изменений размеров (масштаба) исходного объекта достаточно в преобразованных данных разделить координаты на соответствующие собственные значения.

Слежение за объектом на видеопоследовательности

Рассмотрим задачу слежения за объектом на видеопоследовательности. Необходимость в такой задаче возникает, например, при реализации метода дистанционной фотоплетизмографии в палате интенсивной терапии. Здесь в условиях реальной клинической деятельности, где пациент может двигаться, условия освещения изменчивы и действуют другие мешающие факторы, для обеспечения устойчивого формирования сигнала ритмограммы необходимо обеспечивать непрерывное слежение измерительной рамкой в кадре за выбранным участком тела пациента.

Здесь далее описывается практическое решение задачи слежения [108] с использованием программных средств библиотеки *OpenCV**.

В первую очередь для решения задачи необходимо получить фреймы из видеопоследовательности, чтобы провести последующий поиск особых точек и их дескрипторов. Для этого воспользуемся функцией *VideoCapture* библиотеки *OpenCV* для захвата видео. В приводимой ниже программе обеспечивается возможность как обработки видео, находящегося в памяти компьютера, так и изображения с камеры. Также проверяется, присутствует ли видео по заданному адресу расположения файла или возможно ли открыть камеру по заданному идентификационному номеру.

* Работа была выполнена О. А. Головановым, магистрантом УралЭНИН УрФУ.

```

CommandLineParser parser(argc, argv, «{@input_path
|0|input path can be a camera id, like 0,1,2 or a video
filename}»);
    parser.printMessage();
    //считываем расположение файла либо
идентификационный номер камеры
    string input_path = parser.get<string>(0);
    string video_name = input_path;
    VideoCapture video_in;
    if ((isdigit(input_path[0]) && input_path.size() ==
1))
    {
        // начинаем использовать камеру, если передаем
идентификационный номер камеры
        int camera_no = input_path[0] - '0';
        video_in.open(camera_no);
    }
    else {
        //открываем видео файл на 47650 фрейме, если
передаем его расположение на компьютере
        video_in.open(video_name);
        video_in.set(CAP_PROP_POS_FRAMES, 47650);
    }

    //проверка на наличие файла или камеры
    if (!video_in.isOpened()) {
        cerr << «Couldn't open « << video_name <<
endl;
        return 1;
    }

```

Далее при помощи последовательности команд по очереди передаются полученные фреймы из видео или камеры в матрицу *frame* и выводятся на экран:

```

Mat frame;
    // выводим видео на экран
    while (waitKey(1) < 1)
    {

```

```

        video_in >> frame;
        // задаем размер окна для вывода видео
        cv::resizeWindow(video_name, frame.size());
        imshow(video_name, frame);
    }

```

Часто необходимо иметь возможность отслеживать одновременно несколько областей, например, область лба — для определения параметров сердечного ритма — по две области на грудной клетке и в области живота — для определения параметров дыхания. В программе, которая приведена ниже, имеется возможность выбора количества областей для последующего отслеживания. Кроме того, чтобы вычислять особые точки и их дескрипторы только в определенной области, применена функция *selectROI*.

```

//выбор количества областей
int number;
cout << «\nSelect number of districts (from 1 to 5): «;
cin >> number;
for (int i = 0; i < number; i++) {
    // задаём координаты углов прямоугольной области для
    последующего вычисления особых точек и их дескрипторов
    vector<Point2f> bb;
    cv::Rect uBox = cv::selectROI(video_name, frame);
    bb.push_back(cv::Point2f(static_cast<float>(uBox.x),
    static_cast<float>(uBox.y)));
    bb.push_back(cv::Point2f(static_cast<float>(uBox.x +
    uBox.width),
        static_cast<float>(uBox.y)));
    bb.push_back(cv::Point2f(static_cast<float>(uBox.x +
    uBox.width),
        static_cast<float>(uBox.y + uBox.height)));
    bb.push_back(cv::Point2f(static_cast<float>(uBox.x),
    static_cast<float>(uBox.y +
        uBox.height)));
    akaze_tracker.setFirstFrame(frame, bb, «AKAZE»,
    stats, i);
}

```

Приведенный на листинге выше код позволяет выбирать области на определенном фрейме при помощи компьютерной мыши или сенсорной панели. После нажатия на любую клавишу клавиатуры вывод видео на экран прекратится и будет показываться один фрейм, на котором необходимо нажать левой кнопкой и с зажатой левой кнопкой обвести область, на которой впоследствии будут вычислены особые точки и их дескрипторы. Можно заметить, что в цикле присутствует функция *setFirstFrame*, в которую передается само изображение (*frame*), координаты углов вычисляемой области (*bb*), название алгоритма (“AKAZE”), выводимые характеристики (*stats*) и порядковый номер области (*i*).

```
void Tracker::setFirstFrame(const Mat frame,
vector<Point2f> bb, string title, Stats& stats, int k)
{
    // маска необходима для того, чтобы алгоритм работал
    // только в заданной области
    cv::Point *ptMask = new cv::Point[bb.size()];
    const Point* ptContain = { &ptMask[0] };
    int iSize = static_cast<int>(bb.size());
    // считывание координат углов задаваемой области
    for (size_t i = 0; i < bb.size(); i++) {
        ptMask[i].x = static_cast<int>(bb[i].x);
        ptMask[i].y = static_cast<int>(bb[i].y);
    }
    // проверка, необходимая для отрисовки отслеживаемых
    // областей на первом фрейме
    if (k == 0) {
        first_frame = frame.clone();
    }
    cv::Mat matMask = cv::Mat::zeros(frame.size(),
CV_8UC1);
    // заполнение области ограниченной многоугольным
    // контуром для получения маски
    cv::fillPoly(matMask, &ptContain, &iSize, 1,
cv::Scalar::all(255));
    // вычисление координат особых точек first_kp их их
    // дескрипторов first_desc
```

```

    detector->detectAndCompute(first_frame, matMask,
    first_kp[k], first_desc[k]);
    stats.keypoints = (int)first_kp[k].size();
    // отрисовка контуров задаваемой области
    drawBoundingBox(first_frame, bb, 255, 0, 0);
    // выведение текста на изображение
    putText(first_frame, title, Point(0, 60), FONT_
    HERSHEY_PLAIN, 5, Scalar::all(0), 4);
    // запоминание координат области в векторе object_bb
    object_bb[k] = bb;
    delete[] ptMask;
}

```

На рис. 6.20 показан результат работы описанной последовательности действий, где количество измерительных областей, отмеченных красными прямоугольниками, задано равным трем.



Рис 6.20. Области для нахождения особых точек и их дескрипторов на первом фрейме

После вычисления параметров измерительных областей будем отслеживать их на видео или изображении с камеры. Для этого определим основной цикл, который будет останавливаться при условии прекращения поступления фреймов для обработки. В данном цикле будет производиться нахождение координат особых точек и их дескрипторов для каждого последующего фрейма, нахождение совпадений, рисование отслеживаемых областей, вывод на изображение статистики и вывод на экран. Выход из программы осуществляется при помощи нажатия клавиши *Esc*.

```

int i = 0;
for (;;) {
    i++;
    bool update_stats = (i % stats_update_period
== 0);
    video_in >> frame;
    // остановка программы если больше не
поступает фреймов
    if (frame.empty()) break;
    for (int i = 0; i < number; i++) {

        // вычисление координат особых точек и их
дескрипторов
        akaze_res = akaze_tracker.process(frame,
stats, i);
        // обновление статистики и отрисовка её
на фрейме
        akaze_stats += stats;
        if (update_stats) {
            akaze_draw_stats = stats;
        }
        //отрисовка отслеживаемой области и вывод
на экран
        if (i == number - 1) {
            drawStatistics(akaze_res, akaze_
draw_stats);
            cv::resizeWindow(video_name, akaze_
res.size()/2);
            cv::imshow(video_name, akaze_res);
        }
        // организация выхода из программы при
нажатии клавиши Esc
        if (waitKey(1) == 27) break;
    }
}

```

В данном цикле для удобства все вычисления алгоритма вынесены в отдельную функцию *process*, код которой приведен ниже.

В первую очередь находятся особые точки и их дескрипторы для каждого поступающего в нее фрейма, но так как входное изображение имеет достаточно большой размер, область поиска ограничивается для увеличения производительности. С этой целью на первом шаге алгоритма ищутся координаты углов отслеживаемых областей на всем изображении, а в последующих шагах при условии, что вектор *new_bb*, содержащий координаты углов найденных областей не пуст, поиск по всему изображению не производится. Затем с использованием найденных координат задается новая область поиска, которая для надежности слежения выбирается больше, чем отслеживаемая на величину *indent*. Для того чтобы избежать ошибок в функциях *OpenCV*, была осуществлена проверка на выход новой области поиска за границы изображения. Когда отслеживаемая область все-таки теряется, вектор *new_bb* обнуляется и областью поиска снова становится все изображение.

После нахождения координат особых точек в текущей локальной области осуществляется переход к исходной системе координат. С этой целью к каждой координате особой точки добавляются координаты левого верхнего угла области поиска.

```
if (new_bb[k].size() == 0) {
    // при условии, что координаты отслеживаемых
    областей не были найдены, находим особые
    точки на всем изображении
    detector->detectAndCompute(frame, noArray(), kp,
    desc);
    cout << «!»;
}
else {
    // находим координаты новой области поиска
    vector <Point2f> test_bb = new_bb[k];
    double indent = 75;
    for (size_t i = 0; i < new_bb[k].size(); i++) {
        if (i == 0) {
            test_bb[i].x = new_bb[k][i].x - indent;
            test_bb[i].y = new_bb[k][i].y - indent;
        }
        if (i == 1) {
```

```

        test_bb[i].x = new_bb[k][i].x + indent;
        test_bb[i].y = new_bb[k][i].y - indent;
    }
    if (i == 2) {
        test_bb[i].x = new_bb[k][i].x + indent;
        test_bb[i].y = new_bb[k][i].y + indent;
    }
    if (i == 3) {
        test_bb[i].x = new_bb[k][i].x - indent;
        test_bb[i].y = new_bb[k][i].y + indent;
    }
    // проверка на выход за границы изображения
    if (test_bb[i].x > frame.cols) {
        test_bb[i].x = frame.cols;
    }
    else if (test_bb[i].x < 0) {
        test_bb[i].x = 0;
    }
    if (test_bb[i].y > frame.rows) {
        test_bb[i].y = frame.rows;
    }
    else if (test_bb[i].y < 0) {
        test_bb[i].y = 0;
    }
}
// выделение новой области поиска
Rect roi(test_bb[0].x, test_bb[0].y, test_bb[2].x -
test_bb[3].x, test_bb[3].y -
test_bb[0].y);
Mat g_frame = frame(roi);
// запуск алгоритма AKAZE
detector->detectAndCompute(g_frame, noArray(), kp,
desc);

// добавление к координатам особой точки координат
левого верхнего угла области поиска
for (unsigned i = 0; i < kp.size(); i++) {
    kp[i].pt.x += test_bb[0].x;

```

```

        kp[i].pt.y += test_bb[0].y;
    }
    vector <Point2f> push = Points(kp);
    //drawCircle(test_frame, push, 0, 0, 255);
    //drawBoundingBox(test_frame, test_bb, 0, 255, 0);
    // вывод на экран областей поиска
    stringstream ss;
    ss << k+10;
    string name;
    ss >> name;
    cv::imshow(name, g_frame);
    new_bb[k].clear();
}

```

После нахождения координат особых точек и их дескрипторов на первом фрейме и последующем необходимо найти их совпадения. Для этой цели в библиотеке *OpenCV* есть функция *knnMatch*, которая, принимая набор дескрипторов с двух изображений, находит k ближайших совпадений между ними. Это производится с целью последующего нахождения матрицы преобразования одного фрагмента в другой (матрица гомографии).

Библиотека *OpenCV* имеет функцию *findHomography*, которая по списку соответствий возвращает матрицу гомографии. Согласно описанию функции *findHomography* для ее работы нужно как минимум четыре точки, чтобы найти эту матрицу, но предоставление гораздо большего количества точек будет более выгодным, потому что присутствуют шумы и другие несоответствия, влияние которых необходимо минимизировать. Данная матрица состоит из двух основных частей, где первая часть содержит трансформацию сцены, то есть сумму эффектов поворота и перемещения, а вторая часть содержит внутренние параметры камеры. Таким образом, матрица проекционного отображения выглядит следующим образом:

$$H = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \quad (6.45)$$

где c_x, c_y — это главные точки, которые обычно находятся в центре изображения, f_x, f_y — фокусные расстояния, выраженные в единицах пикселей, r — матрица поворотов, t — перемещение точки.

При вычислении матрицы гомографии (см. листинг ниже) используется метод *RANSAC* [120], также известный как метод «случайной выборки с консенсусом» (“random sampling with consensus”). В этом методе все множество точек разбивается на подмножества случайным образом, и матрица гомографии вычисляется только для одного подмножества. Затем она уточняется по всем оставшимся подмножествам, которые примерно согласуются с первоначальной оценкой. Алгоритм вычисляет множество таких случайных выборок и сохраняет ту, которая имеет наибольшую долю вложений для вычисления матрицы. Метод чрезвычайно эффективен для избавления от шумов и поиска «правильных» особых точек.

Одним из элементов, входящих в функцию *findHomography*, является матрица *inlier_mask*, которая является выходной. Эта матрица заполняется массивом значений с указанием того, какие точки были использованы при наилучшем вычислении матрицы гомографии. Полученная матрица применяется к точкам границы отслеживаемой области из первого фрейма, для этого используется функция *perspectiveTransform*.

```
// нахождение совпадений особых точек между двумя
изображениями
matcher->knnMatch(first_desc[k], desc, matches, 2);
for (unsigned i = 0; i < matches.size(); i++)
{
    matched1.push_back(first_kp[k][matches[i]
[0].queryIdx]);
    matched2.push_back(kp[matches[i][0].
trainIdx]);
}
//вычисление матрицы гомографии
Mat inlier_mask, homography;
vector<KeyPoint> inliers1, inliers2;
if (matched1.size() >= 4) {
    homography = findHomography(Points(match
ed1), Points(matched2),
```

```

        RANSAC, ransac_thresh, inlier_mask);
    }
    // нахождение новых координат отслеживаемой
    области
    perspectiveTransform(object_bb[k], new_bb[k],
    homography);
    // сохраняем выбранные при вычислении
    гомографии точки
    for (unsigned i = 0; i < matched1.size(); i++)
    {
        if (inlier_mask.at<uchar>(i)) {
            int new_i = static_
cast<int>(inliers1.size());
            inliers1.push_back(matched1[i]);
            inliers2.push_back(matched2[i]);
        }
    }
}

```

Для рассматриваемой измерительной задачи требуется, чтобы отслеживаемая область не изменяла своих размеров с целью более точного вычисления параметров дыхания и сердечного ритма. Для этого, исходя из изначальных высоты и ширины области, а также угла наклона, который был получен в результате выполнения описанной выше последовательности действий, на листинге ниже вычисляются новые координаты, которые сохраняют размер области.

```

// вычисление высоты и ширины отслеживаемой области
double A = length(object_bb[k][3].x, object_bb[k][3].y,
object_bb[k][2].x, object_bb[k][2].y);
double B = length(object_bb[k][3].x, object_bb[k][3].y,
object_bb[k][0].x, object_bb[k][0].y);
//вычисление угла наклона отслеживаемой области
double alpha = -(new_bb[k][3].y - new_bb[k][2].y) /
(new_bb[k][2].x - new_bb[k][3].x);
new_bb[k][2].x = (A)*cos(alpha) - (0)*sin(alpha) + new_
bb[k][3].x;
new_bb[k][2].y = (A)*sin(alpha) + (0)*cos(alpha) + new_
bb[k][3].y;

```

```

new_bb[k][0].x = (0)*cos(alpha) - (-B)*sin(alpha) +
new_bb[k][3].x;
new_bb[k][0].y = (0)*sin(alpha) + (-B)*cos(alpha) +
new_bb[k][3].y;
new_bb[k][1].x = (A)*cos(alpha) - (-B)*sin(alpha) +
new_bb[k][3].x;
new_bb[k][1].y = (A)*sin(alpha) + (-B)*cos(alpha) +
new_bb[k][3].y;

```

Практическое задание к п. 6.5

Выполните обработку видеопоследовательности лица с веб-камеры:

- 1) воспроизведите описанную выше реализацию программы и проведите ее испытания. Целью испытаний является проверка работоспособности метода слежения при изменении освещения, при расфокусировке камеры и при перемещении области отслеживания;
- 2) выполните оценку характеристик метода слежения видеопоследовательности, а именно количество кадров в секунду и отношение «правильных» совпадений особых точек ко всем обнаруженным совпадениям;
- 3) реализуйте метод Лукаса-Канаде с использованием библиотеки *OpenCV* для видеопоследовательности лица;
- 4) реализуйте метод главных компонент для обработки видеопоследовательности лица с веб-камеры.

6.6. Анализ видеоизображения для оценки сердечного ритма

В качестве примера использования анализа видеопоследовательности в этом пункте описывается метод дистанционной фотоплетизмографии для определения вариабельности сердечного ритма*.

Измерение жизненно важных медицинских признаков, таких как частота сердечных сокращений, с помощью фотоплетизмографии

* Материал подготовлен при участии Д. О. Коноваловой, магистра УралЭНИН УрФУ.

было начато в ранних исследованиях Херцмана по кровообращению в конечностях. В настоящее время для измерения пульса широко используются пульсовые оксиметры. Эти измерения основаны на контактной фотоплетизмографии. Несмотря на успех этого метода, современные фотоплетизмографы не могут быть использованы в ситуациях с чувствительной и поврежденной кожей или в случае, когда измерение должно проходить бесконтактно. Поэтому в последнее время предпринимаются попытки к реализации дистанционной фотоплетизмографии. Известны исследования, в которых показано, что анализ видео с камеры приводит к результатам, аналогичным фотоплетизмографии [121].

Алгоритм формирования сигнала дистанционной фотоплетизмографии

Для бесконтактного определения пульса по видеоизображению будем основываться на анализе изменения интенсивности цвета кожи на видеоизображении. Известно, что пульсовая волна для фотоплетизмографии имеет наилучшее соотношение сигнала и шума в области лба. Поэтому для получения сигнала будем формировать зону интереса именно в этой области.

Обработка кадров и выделение интересующего участка для измерения

Определить положение лица на видеоизображении, т. е. найти положение и размер прямоугольной рамки лица, можно методом Виолы-Джонса [122], который реализован в библиотеке *OpenCV* [108]. С целью увеличения скорости обработки для локализации лица используется полутоновое преобразование цветного изображения. Далее выделяется интересующая зона лба. Размеры области лба для простоты составляют 20 % от высоты и 30 % ширины детектируемого прямоугольника лица. Координаты центра зоны интереса составляют 50 % от ширины прямоугольника и 15 % от высоты. Также для получения более точного результата необходимо в измерительном прямоугольнике исключить области, где нет открытых участков кожи. Для этого выделенный фрагмент переводится в формат HSV [98] и удаляются области, которые не попадают в выбранный цветовой тон. Оставшаяся часть фрагмента в дальнейшем использует-

ся для получения сигнала ВСП. При обработке сигнала применяется метод главных компонент, который был описан выше в пункте 6.5. Для использования метода главных компонент выделенная зона интереса разделяется на n (в рассматриваемом примере $n = 6$) парциальных участков, в каждом из которых формируется парциальный сигнал согласно алгоритму, описанному ниже.

Алгоритм формирования и первичной обработки исходного сигнала

Как уже было сказано, исходный сигнал формируется путем пространственного усреднения яркости пикселей в зеленом канале каждого из n парциальных участков.

После получения сигнала производится его нормировка по следующей формуле:

$$x_{norm} = \frac{x_i - \bar{x}}{\sigma^i},$$

где x_i — среднее значение амплитуды элементов парциального сигнала, σ^i — среднеквадратическое отклонение амплитуды элементов парциального сигнала.

На рис. 6.21 показан пример фрагмента исходного сигнала, а на рисунке 6.22 — тот же фрагмент после нормализации.



Рис. 6.21. Исходный сигнал в зоне интереса, сформированный из усреднения яркости пикселей в зеленом канале



Рис. 6.22. Нормализованный сигнал в зоне интереса

Более точно алгоритм первичной обработки исходного сигнала описывается следующим образом.

1. Записываем полученные парциальные сигналы в виде матрицы:

$$X = \begin{pmatrix} x_{1,1} & \cdots & x_{1,N} \\ \vdots & \ddots & \vdots \\ x_{n,1} & \cdots & x_{n,N} \end{pmatrix},$$

где n — количество парциальных сигналов, N — количество элементов в каждом парциальном сигнале.

2. Рассчитываем матрицу средних значений M , каждый элемент которой M_k рассчитывается по формуле:

$$M_k = \frac{1}{N} \sum_{i=1}^N x_{k,i}, k = 1, \dots, n.$$

3. Определяем матрицу ковариации P :

$$P = \frac{1}{N-1} ((X - N)(X - N)^T).$$

4. Для полученной матрицы ковариации рассчитываем собственные значения λ_i из условия равенства нулю детерминанта:

$$|P - \lambda_i E| = 0.$$

5. Для каждого собственного значения λ_i рассчитываем собственный вектор $\vec{v}^i$:

$$\vec{v}^i (S - \lambda_i E) = 0.$$

6. Производим сортировку векторов $\vec{v}^i$ в порядке возрастания λ_i . При этом выбираем только j первых векторов $\vec{v}^i$, соответствующих наибольшему λ_i , а значение j не должно превышать n . В данном случае достаточно взять $j=1$. Обозначаем полученную матрицу собственных векторов как

$$W = \begin{pmatrix} v_1^1 & \cdots & v_n^1 \\ \vdots & \ddots & \vdots \\ v_1^j & \cdots & v_n^j \end{pmatrix}.$$

7. Производим ортогональное преобразование парциальных сигналов по формуле

$$Y = WX.$$

Каждая строка матрицы Y соответствует значениям, полученным при помощи векторов $\vec{v}^i$, отсортированных в порядке возрастания собственных чисел (в нашем случае матрица Y содержит одну строку данных).

8. Производим фильтрацию полученного сигнала с помощью полосового фильтра Баттерворда 5-го порядка (см. пункт 6.3) для области частот от 0,667 до 2,5 Гц, в пределах которой изменяется частота сердечного ритма.

В результате будет получен сигнал, из которого впоследствии будет сформирована временная последовательность ритмограммы, известная в литературе как *rPPG (remote photoplethysmography)*.

Построение временных диаграмм

Для того чтобы получить сигнал *rPPG*, необходимо построить временную диаграмму. Временная диаграмма измеренного сигнала — это последовательность величин временных интервалов между пиками исходного сигнала, приведенная к равномерной шкале текущего времени.

Пики сигнала *rPPG* считаются как локальный экстремум по заданному окну в 11 отсчетов. Это количество отсчетов было выбрано исходя из того, что максимальная частота 2,5 Гц, а у видео частота ка-

дров — 29 Гц, следовательно, максимально возможное окно составляет 11 отсчетов.

Необходимо также учитывать, что пики, полученные с сигнала ЭКГ и $rPPG$, не будут совпадать, т. к. у сигнала $rPPG$ от сигнала ЭКГ есть задержка.

На рис. 6.23, 6.24 представлены сигналы ЭКГ и $rPPG$, с выделенными пиками.

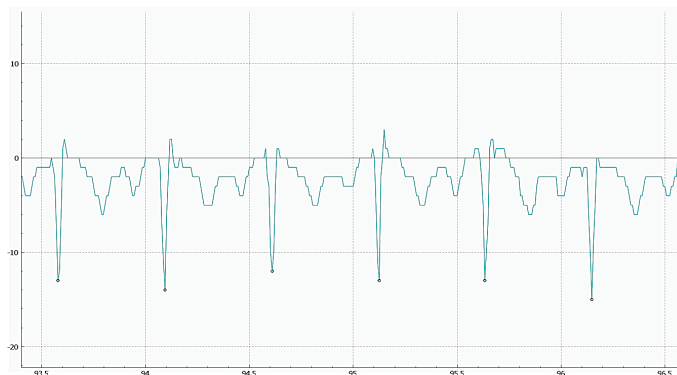


Рис. 6.23. Получение пиков ВСП ЭКГ

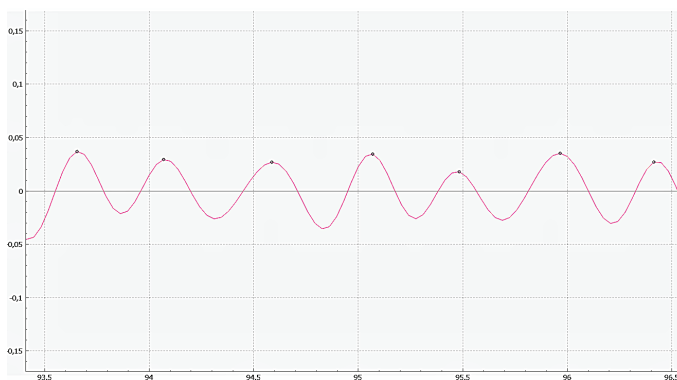


Рис. 6.24. Получение пиков ВСП $rPPG$

Для построения временных диаграмм можно использовать следующий алгоритм:

- 1) выделяем интересующие нас точки экстремумов (t_i, y_i), где y_i — амплитуда сигнала в момент t_i (рис. 6.25). В рассматриваемом случае они должны соответствовать пикам исходного сигнала. Найденные точки образуют временную последовательность

$(t_1, y_1), (t_2, y_2), \dots, (t_n, y_n)$, где n — количество найденных пиков:
 $t_1 < t_2 < \dots < t_{n-1} < t_n$;

- 2) формируем временную диаграмму с неравномерной шкалой. Она состоит из следующих пар чисел: $(t_1, t_2 - t_1), (t_2, t_3 - t_2), \dots, (t_{n-1}, t_n - t_{n-1})$ (рис. 6.26);
- 3) преобразуем неравномерную шкалу к равномерной посекундной шкале методом линейной интерполяции (рис. 6.27).

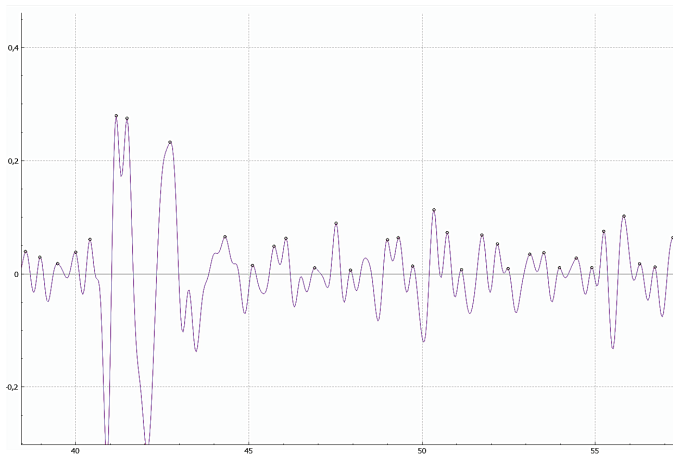


Рис. 6.25. Исходный сигнал для построения временной диаграммы и выделенные экстремальные точки

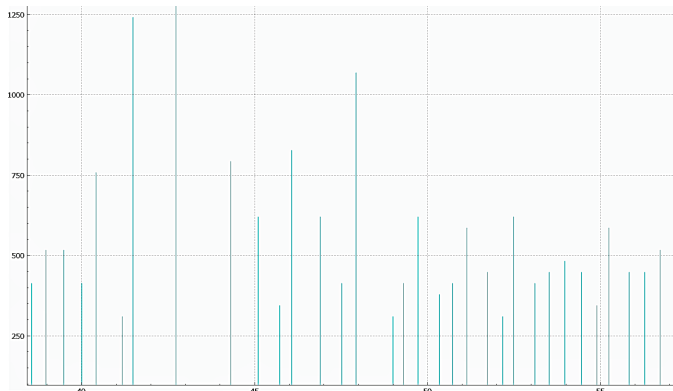


Рис. 6.26. Временная диаграмма с неравномерной шкалой, где положение вертикальных отрезков соответствует выделенным точкам на графике предыдущего рисунка, а высота отрезков равна временным интервалам между точкой положения отрезка и следующей за ней выделенной точкой

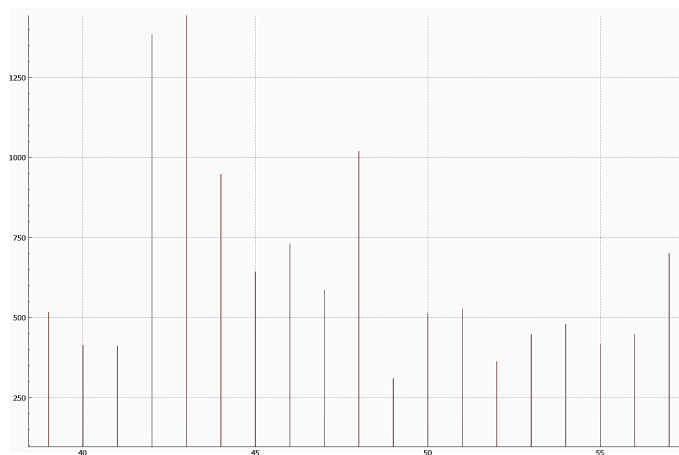


Рис. 6.27. Временная диаграмма с равномерной шкалой, полученная с помощью линейной интерполяции диаграммы, показанной на рис. 6.26

Методы сравнения временных рядов ВСП на примере сравнения ритмограммы *rPPG* и кардиоинтервалограммы ЭКГ

Поскольку биофизическая природа сигналов ЭКГ и *rPPG* различна, то и сигналы не обязаны совпадать или даже быть близкими по амплитуде. Учитывая, что анализ вариабельности сердечного ритма по данным ЭКГ основывается на измерении длительности сердечного цикла (R - R интервал), можно предположить, что анализ вариабельности по данным *rPPG* основывается на измерении длительности пульсовой волны (ДПВ). В стационарных условиях при неизменных характеристиках свойств сосудов величины R - R интервала и ДПВ должны полностью совпадать. Получение данных при помощи *rPPG* приводит к некоторой задержке, это связано с тем, что необходимо время, чтобы кровь ударного объема достигла места, где производится считывание сигнала. При изменении свойств сосудов такая задержка может увеличиваться или уменьшаться. Очевидно, что свойства сосудов зависят от физических свойств и особенностей организма, поэтому сигнал *rPPG* может изменяться.

В исследованиях [123; 124] описана взаимосвязь вариабельности сигналов ЭКГ и *rPPG* и предложены методы, которые можно использовать для сопоставления вариабельности положения их пиков.

Сравнение спектров в целом и в поддиапазонах

При исследовании сигналов часто используются методы спектрального анализа, которые позволяют получить численные оценки частотного состава сигнала. Наиболее распространены методики спектрального анализа, основанные на дискретном преобразовании Фурье.

На рис. 6.28 представлены спектры ритмокардиограмм ЭКГ и *rPPG* по всем диапазонам частот. Как видно из рис. 6.28, спектры по всем диапазонам частот сложно сравнивать. Поэтому сравнение лучше проводить по поддиапазнам. На рис. 6.29–6.31 представлены спектры ритмокардиограмм, фильтрованные по стандартным для ВСР диапазонам частот 0,003–0,04 Гц (диапазон *VLF*), 0,04–0,15 Гц (диапазон *LF*), 0,15–0,4 Гц (диапазон *HF*) соответственно. Можно заметить, что огибающие спектры ритмограмм ЭКГ и *rPPG* в отдельных диапазонах частот имеют качественное сходство.

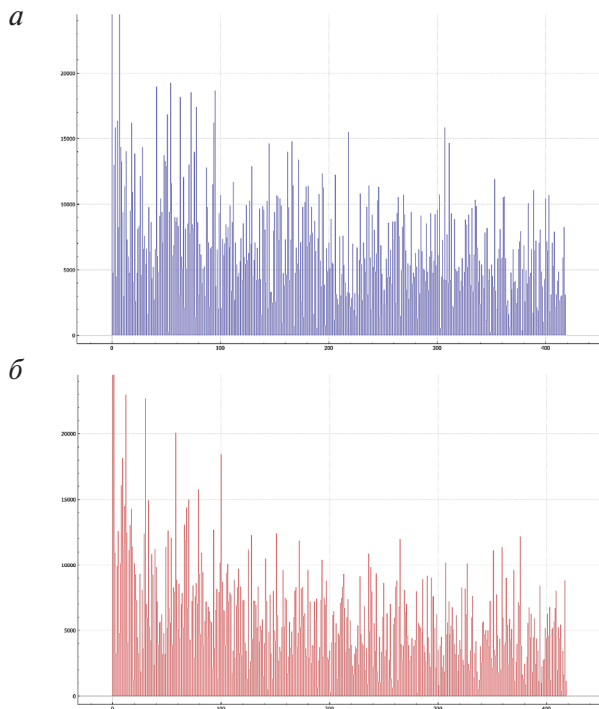


Рис. 6.28. Спектры временных рядов по всем диапазонам частот:
а — спектр временных рядов ЭКГ; *б* — спектр временных рядов *rPPG*

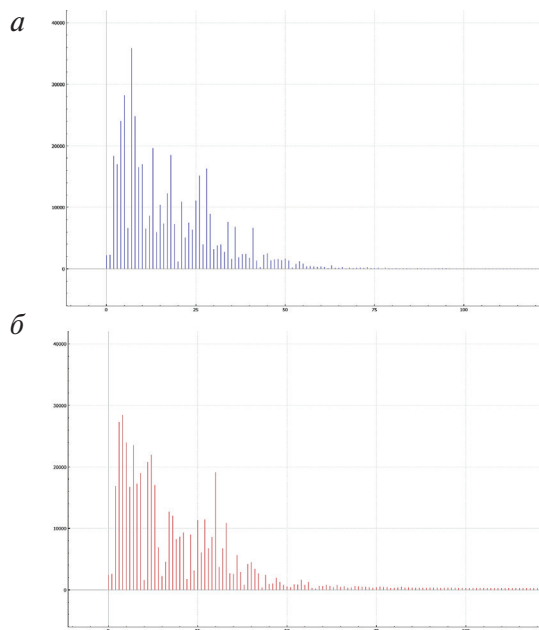


Рис. 6.29. Спектры временных рядов в диапазоне частот VLF 0,003–0,04 Гц:
 а — спектр временных рядов ЭКГ; б — спектр временных рядов $rPPG$

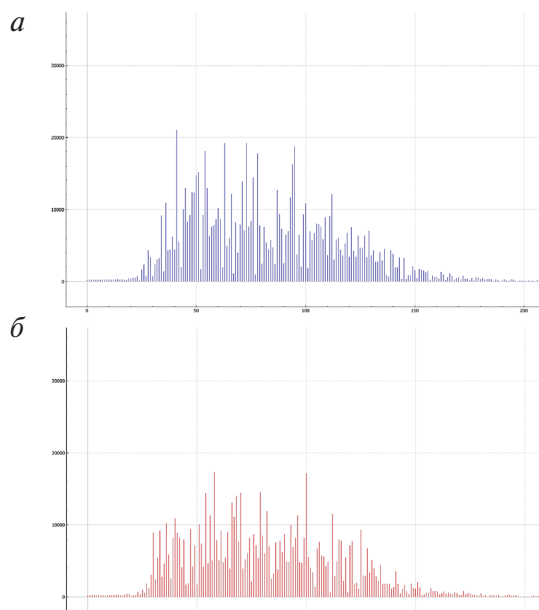


Рис. 6.30. Спектры временных рядов в диапазоне частот LF 0,04–0,15 Гц:
 а — спектр временных рядов ЭКГ; б — спектр временных рядов $rPPG$

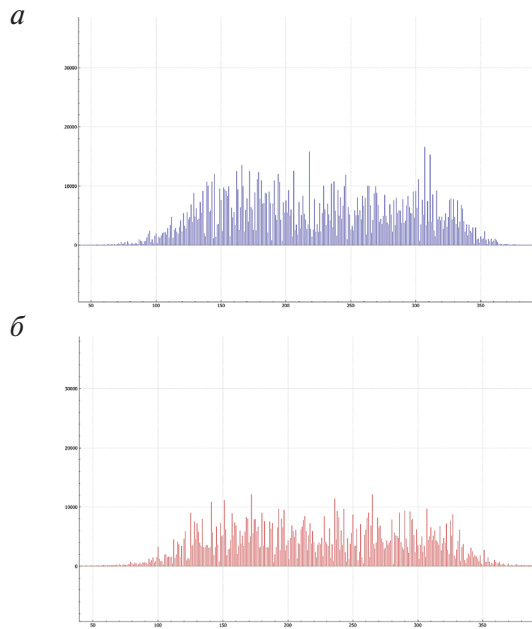


Рис. 6.31. Спектры временных рядов в диапазоне частот HF 0,15–0,4 Гц:

а — спектр временных рядов ЭКГ; *б* — спектр временных рядов *rPPG*

Сравнение гистограмм в разных диапазонах частот

Одним из способов сопоставления временных рядов является анализ гистограмм амплитуд этих рядов. На рис. 6.32–6.35 представлены гистограммы значений временных рядов в различных диапазонах частот. На рис. 6.32 можно видеть, что частота пиков *rPPG* составляет примерно 0,519 Гц, что соответствует примерно 115 ударам в мин. Видно, что эти значения совпадают с данными ЭКГ. Можно видеть, что гистограммы в поддиапазонах имеют еще большее сходство.

Оценка взаимной корреляционной функции ритмограмм

При анализе сигналов также широко применяется функция взаимной корреляции (ФВК). Она позволяет судить об общих периодичностях в сигналах. При анализе медико-биологических сигналов часто применяется коэффициент взаимной корреляции (КВК) [124]. Оценки КВК для двух дискретных сигналов рассчитываются по формуле:

$$R_{xy} = \frac{1}{N\sigma_x\sigma_y} \sum_{n=0}^{N-1} x(n)y(n),$$

где $x(n), y(n)$ — отсчеты сигнала, N — число отсчетов, σ_x, σ_y — оценка среднеквадратических отклонений этих сигналов соответственно.

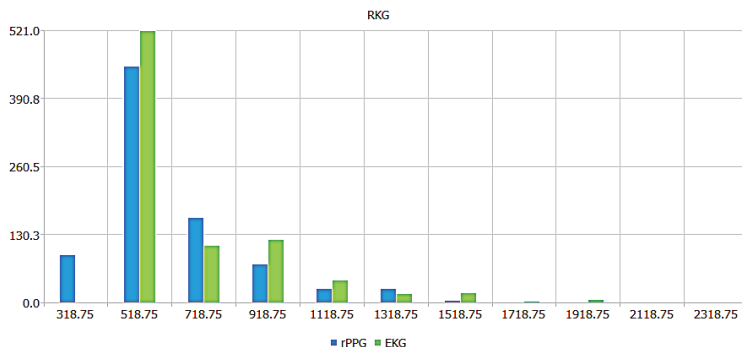


Рис. 6.32. Гистограммы временных рядов по всем диапазонам частот

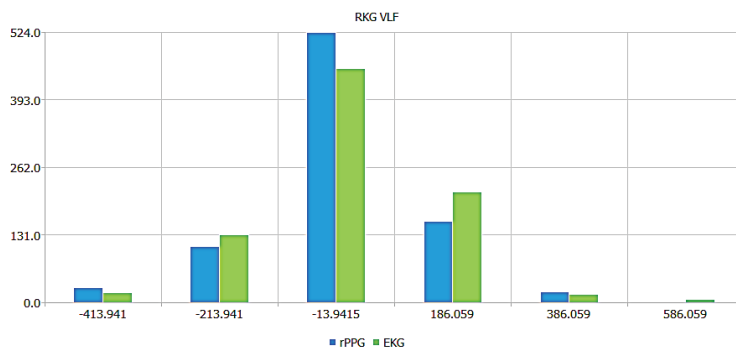


Рис. 6.33. Гистограммы временных рядов по диапазону VLF 0,003–0,04 Гц

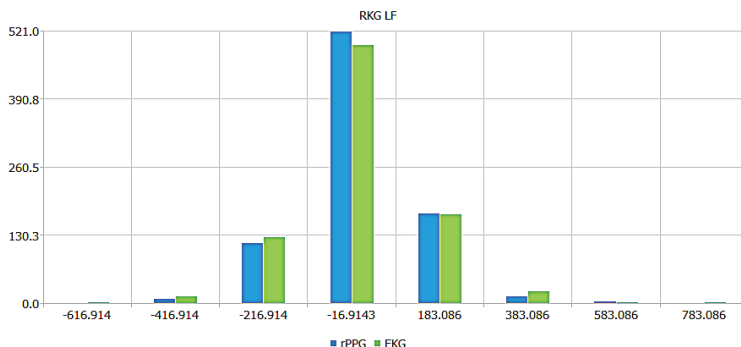


Рис. 6.34. Гистограммы временных рядов по диапазону LF 0,04–0,15 Гц

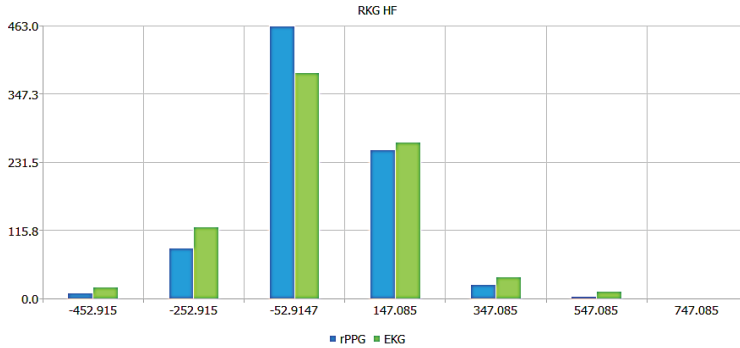


Рис. 6.35. Гистограммы временных рядов по диапазону HF 0,15–0,4 Гц

Для расчета ФВК будем считать КВК в зависимости от величины сдвига относительно и наоборот. Суммирование будем производить окном, размер которого равен половине количества отсчетов. Исходя из этого, выражение, с помощью которого вычисляется ФВК, имеет следующий вид:

$$R_{xy}(m) = \begin{cases} \frac{1}{N\sigma_x\sigma_y} \sum_{n=0}^{\frac{1}{2}N-1} x(n)y(n+m), m=0,1,\dots,\frac{1}{2}N-1 \\ \frac{1}{N\sigma_x\sigma_y} \sum_{n=0}^{\frac{1}{2}N-1} x(n-m)y(n), m=-1,-2,\dots,-\left(\frac{1}{2}N-1\right) \end{cases},$$

где m — аргумент ФВК, равный сдвигу сигналов относительно друг друга.

Значения ФВК лежат в пределах от -1 до $+1$. Функция позволяет оценить наличие схожих по частоте колебаний в двух сигналах. Если при каких-то значениях сдвига модуль значения ФВК будет приближаться к единице, это будет служить признаком наличия в сигналах однотипных изменений, которые происходят с задержкой, равной сдвигу.

На рис. 6.36 представлена ФВК для поддиапазона VLF . Ось абсцисс соответствует числу сдвигов, ось ординат — коэффициенту взаимной корреляции.

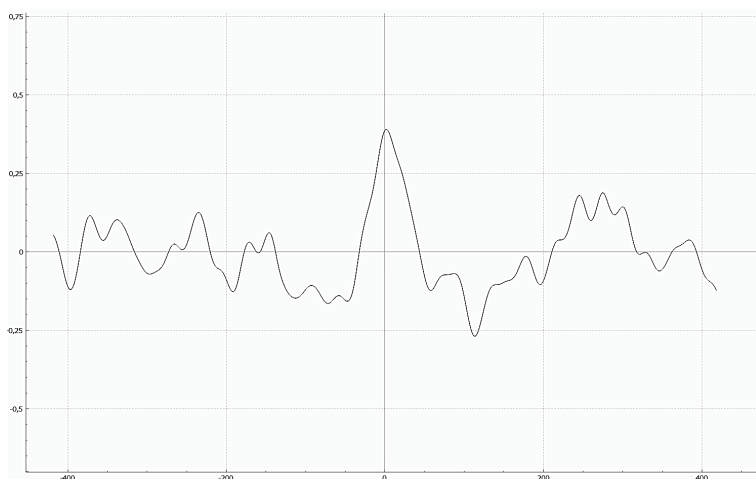


Рис. 6.36. ФВК временных рядов ЭКГ и *rPPG* в диапазоне частот *VLF* 0,003–0,04 Гц

Практическое задание к п. 6.6

Выполните анализ видеопоследовательности для оценки сердечного ритма:

- 1) реализуйте выделение некоторого участка тела испытуемого на видеопоследовательности, сформируйте и обработайте сигнал с помощью метода главных компонент и фильтра Баттерворда;
- 2) разработайте программную реализацию задачи 1 для формирования временных диаграмм.

Контрольные вопросы к главе 6

1. Перечислите базовые операции над изображениями. Опишите алгоритмы эквализации гистограммы яркости и пороговой обработки.
2. Перечислите простые и совершенные методы обнаружения контуров.
3. Перечислите методы улучшения изображений. Опишите алгоритмы сглаживания и повышения резкости.

4. Опишите алгоритмы восстановления изображений. Сравните принципы работы фильтра Винера и фильтрации по Тихонову.
5. Перечислите дескрипторы особенностей. Сравните достоинства и недостатки алгоритмов *SIFT*, *SURF*, *ORB*, *AKAZE*.
6. Охарактеризуйте обработку видеопоследовательности. Сформулируйте понятие оптического потока.
7. Опишите алгоритм анализа видеоизображения для оценки сердечного ритма.

Список библиографических ссылок

1. 65+ in the United States: 2010 / L. A. West [et al.] // US Census Bureau. 2014. P. 23–212.
2. World Health Organization. Strategizing national health in the 21st century: a handbook / World Health Organization, 2016. URL: <https://www.who.int/publications-detail/strategizing-national-health-in-the-21st-century-a-handbook> (дата обращения: 17.09.2019).
3. Журавлев Д. А., Сорокина Л. А. Лечить болезнь или больного? // Артериальная гипертензия. 2010. Т. 16. № 3. С. 343–345.
4. Reappraisal of European guidelines on hypertension management: a European Society of Hypertension Task Force document / G. Mancia [et al.] // Blood pressure. 2009. Vol. 18. № 6. P. 308–347.
5. Судаков К. В. Теория функциональных систем и ее применение в физиологии и медицине // Новости медико-биологических наук. 2004. № 4. С. 109–133.
6. Агаджанян А. Н., Смирнов В. М. Нормальная физиология: учебник для студентов медицинских вузов М. : Медицинское информационное агентство, 2009. 520 с.
7. Анохин П. К. Принципиальные вопросы общей теории функциональных систем // Принципы системной организации функций. М. : Наука. 1973. С. 5–61.
8. Анохин П. К. Системные механизмы высшей нервной деятельности: избранные труды. М. : Наука, 1979. 454 с.
9. Ткаченко Б. И. Нормальная физиология человека. М. : Медицина. 2005. С. 928.
10. Torday J. Homeostasis as the mechanism of evolution // Biology. 2015. Vol. 4. № 3. P. 573–590.
11. A physiologist's view of homeostasis / H. Modell [et al.] // Advances in physiology education. 2015. Vol. 39. № 4. P. 259–266.

12. Шмидт Р., Тевс Г. Физиология человека. В 3-х т. Т. 2. М. : Мир, 1996. 313 с.
13. Общий курс физиологии человека и животных. В 2 кн. Кн. 1 / А. Д. Ноздрачев, Ю. И. Баженов, И. А. Баранникова. М. : Высшая школа, 1991. 509 с.
14. Бабский Е. Б., Косицкий Г. И., Ходоров Б. И. Физиология человека / М. : Медицина, 1985. 544 с.
15. Гершел Р. Секреты физиологии // М., СПб. : Бином : Невский диалект, 2001. 448 с.
16. Стресс без дистресса / Г. Селье [и др.]. М. : Прогресс, 1979. 126 с.
17. ECG-based heartbeat classification for arrhythmia detection: A survey / E. J. da S. Luz [et al.] // Computer methods and programs in biomedicine. 2016. Vol. 127. P. 144–164.
18. Applications of artificial neural networks for ECG signal detection and classification / Y. H. Hu [et al.] // Journal of electrocardiology. 1993. Vol. 26. P. 66–73.
19. Poli R., Cagnoni S., Valli G. Genetic design of optimum linear and nonlinear QRS detectors // IEEE Transactions on Biomedical Engineering. 1995. Vol. 42. № 11. P. 1137–1141.
20. Wavelet transform-based QRS complex detector / S. Kadambe, R. Murray, G. F. Boudreaux-Bartels // IEEE Transactions on biomedical Engineering. 1999. Vol. 46. № 7. P. 838–848.
21. ECG beat detection using filter banks / V. X. Afonso [et al.] // IEEE transactions on biomedical engineering. 1999. Vol. 46. № 2. P. 192–202.
22. Kohler B. U., Hennig C., Orglmeister R. QRS detection using zero crossing counts // Applied genomics and proteomics. 2003. Vol. 2. № 2. 138–145.
23. Suppappola S., Sun Y. Nonlinear transforms of ECG signals for digital QRS detection: a quantitative analysis // IEEE Transactions on Biomedical Engineering. 1994. Vol. 41. № 4. P. 397–400.
24. Trahanias P. E. An approach to QRS complex detection using mathematical morphology // IEEE Transactions on Biomedical Engineering. 1993. Vol. 40. № 2. P. 201–205.
25. Истомин Б. А. Система и алгоритм обнаружения информативных импульсов электрокардиосигнала в условиях интенсивных помех. Пенза: Пензенский государственный университет, 2011. 22 с.

26. Pan J., Tompkins W. J. A real-time QRS detection algorithm // IEEE transactions on biomedical engineering. 1985. № 3. P. 230–236.
27. Баевский Р. М., Иванов Г. Г., Чирейкин Л. В. Методические рекомендации: Анализ вариабельности сердечного ритма при использовании различных электрокардиографических систем // Вестник аритмологии. 2001. № 24. С. 65–87.
28. Баевский Р. М., Береснева А. П. Введение в донозологическую диагностику. М. : Фирма «Слово», 2008. 220 с.
29. Analysis of heart rate variability in hypertensive patients before and after treatment with angiotensin II–converting enzyme inhibitors / M. Júnior [et al.] // Arquivos Brasileiros de Cardiologia. 2004. Vol. 83. № 2. P. 165–168.
30. Вариабельность сердечного ритма на различных этапах гестационного процесса / С. Л. Дмитриева [и др.] // Научно-методический электронный журнал Концепт. 2013. 124 с.
31. Decomposing skin conductance into tonic and phasic components / C. L. Lim [et al.] // International Journal of Psychophysiology. 1997. Vol. 25. № 2. P. 97–109.
32. Мясникова Н. Е., Проскуряков Г. М. Медико-биологические основы работы стабилметрических платформ // Анализ, синтез и управление в сложных системах: сб. науч. трудов. 2009. С. 25–23.
33. Кубряк О. В., Гороховский С. С. Практическая стабилметрия. Статические двигательные-когнитивные тесты с биологической обратной связью по опорной реакции. М. : Маска, 2012. С. 88.
34. Программно-методическое обеспечение компьютерного стабиллографического комплекса с биологической обратной связью «Стабилан-01-2» / ЗАО «ОКБ «РИТМ». Таганрог, 2011. 279 с.
35. Скворцов Д. В. Диагностика двигательной патологии инструментальными методами: анализ походки, стабилметрия. М. : науч.-мед. фирма «МБН», 2007. 617 с.
36. Физиологические методы контроля в спорте. Контроль равновесия и координационных способностей методом стабиллографии / Л. В. Капилевич [и др.]. Томск: Изд-во Томского политех. ун-та, 2009. 172 с.
37. Гаже П. М. Постурология. Регуляция и нарушения равновесия тела человека. СПб. : Издательский дом СПбМАПО, 2008. 312 с.

38. Болобан В. Н., Литвиненко Ю. В., Нижниковски Т. Системная стабильнография: методология и методы измерения, анализа и оценки статодинамической устойчивости тела спортсмена и системы тел // Наука в олимпийском спорте. 2012. № 1. С. 27–35.
39. Зинурова Н. Г. Особенности функционального состояния сердечно-сосудистой системы спортсменов с различной направленностью физических нагрузок и статокINETической устойчивостью : автореферат дис. ... канд. биол. наук : 03.03.01 / Зинурова Надежда Григорьевна; [Место защиты: Челяб. гос. пед. ун-т]. Челябинск, 2015. 24 с.
40. Назаренко А. С., Мавлиев Ф. С., Хаснутдинов Н. Ш. Функция равновесия тела в условиях утомления мышц плечевого пояса у спортсменов разных видов спорта // Ученые записки университета имени П. Ф. Лесгафта. 2015. Т. 123. № 5.
41. Акжигитов Р. Ф. Перспективы применения комплекса «Стабилян» для тестирования спортсменов // Известия Южного федерального университета. Технические науки. 2010. Т. 109. № 8. С. 8–12.
42. Лебедева Н. Н., Каримова Е. Д. Акустические характеристики речевого сигнала как показатель функционального состояния человека // Успехи физиологических наук. 2014. Т. 1. № 45. С. 57–95.
43. Hadjileontiadis L. J. Biosignals and compression standards // M-Health. Springer. 2006. P. 277–292.
44. StorageBIT-A Metadata-aware, Extensible, Semantic and Hierarchical Database for Biosignals / C. Carreiras [et al.] // HEALTHINF. 2013. P. 65–74.
45. BioSignalML – a meta-model for biosignals / D. J. Brooks [et al.] // 2011 annual international conference of the IEEE engineering in medicine and biology society. IEEE. 2011. P. 5670–5673.
46. EDF(+) programming guidelines. URL: <https://www.edfplus.info/specs/guidelines.html> (дата обращения: 13.09.2019).
47. Biosemi EEG ECG EMG BSPM NEURO amplifiers systems. URL: https://www.biosemi.com/faq/file_format.htm (дата обращения: 16.09.2019).
48. Schlögl A. GDF-a general dataformat for biosignals // arXiv preprint cs/0608052 (дата обращения: 16.09.2019).

49. GDF-A general dataformat for biosignals Version 1.25 / A. Schlögl [et al.]. Technical report, Institute of Biomedical Engineering, University of ..., 2005. URL: https://www.researchgate.net/publication/250353054_GDF_-_A_GENERAL_DATAFORMAT_FOR_BIOSIGNALS_VERSION_125 (дата обращения 19.09.2019).
50. HDF5 File Format Specification Version 3.0. URL: <https://portal.hdfgroup.org/display/HDF5/File+Format+Specification> (дата обращения: 13.09.2019).
51. MATLAB. version 9.7 (R2019a) / MATLAB. Natick, Massachusetts: The MathWorks Inc., 2019. URL: <https://1progs.ru/matlab/> (дата обращения: 19.09.2019).
52. PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals / A. L. Goldberger [et al.] // *Circulation*. 2000. Vol. 101. № 23. P. 215–220.
53. Schlögl A. An overview on data formats for biomedical signals // *World Congress on Medical Physics and Biomedical Engineering*, September 7–12, 2009, Munich, Germany // Springer. 2009. P. 1557–1560.
54. BIOPAC Systems, Inc. ACKAPI BIOPAC FILE FORMAT API. URL: <https://www.biopac.com/wp-content/uploads/ACKAPI.pdf>. (дата обращения: 13.09.2019).
55. TDMS File Format Internal Structure – National Instruments. URL: <http://www.ni.com/product-documentation/5696/en/> (дата обращения: 16.09.2019).
56. C3D.ORG – The biomechanics standard file format. URL: <https://www.c3d.org/> (дата обращения: 16.09.2019).
57. About MFER. URL: <http://www.mfer.org/en/WhatIsMFER.htm> (дата обращения: 16.09.2019).
58. The UCR time series archive / H. A. Dau [et al.] // *arXiv preprint arXiv:1810.07758*. 2018. URL: <https://arxiv.org/abs/1810.07758> (дата обращения: 16.09.2019).
59. Asuncion A., Newman D. UCI machine learning repository. Irvine, CA: University of California // School of Information and Computer Science. *Intell.* Vol. 137, № 1–2.
60. Deap: A database for emotion analysis; using physiological signals / S. Koelstra [et al.] // *IEEE transactions on affective computing*. 2011. Vol. 3. № 1. P. 18–31.

61. The Cardiac Safety Research Consortium electrocardiogram warehouse: thorough QT database specifications and principles of use for algorithm development and testing / P. Kligfield [et al.] // American heart journal. 2010. Vol. 160. № 6. P. 1023–1028.
62. TonyBagnall / time-series-classification – Bitbucket. URL: <https://bitbucket.org/TonyBagnall/time-series-classification/src/default/> (дата обращения: 01.11.2019).
63. PIA-Group/BioSPPy. URL: <https://github.com/PIA-Group/BioSPPy> (дата обращения: 01.11.2019).
64. Notebook – biosignalsnotebooks. URL: https://www.biosignalsplux.com/notebooks/Categories/MainFiles/biosignalsnotebooks_rev.php (дата обращения: 01.11.2019).
65. Vidaurre C., Sander T. H., Schlögl A. Vidaurre C. BioSig: The Free and Open Source Software Library for Biomedical Signal Processing // Computational Intelligence and Neuroscience. 2011. Vol. 2011. BioSig.
66. Brunner C. URL: <https://github.com/cbrnr/sigviewer> (дата обращения: 01.11.2019).
67. Рангайян Р. М. Анализ биомедицинских сигналов. Практический подход / пер. с англ. А. Н. Калиниченко; под ред. А. П. Немирко. М. : Физматлит, 2010. 440 с.
68. Михайлов В. М. Вариабельность ритма сердца. Опыт практического применения метода. Иваново, 2000. 200 с.
69. Бендат Д., Пирсол А. Прикладной анализ случайных данных. М. : Мир, 1989. 540 с.
70. Зайцев В.К., Кисилев В. А. Методика вариационной пульсометрии // Медицинские новости. 2010. № 7. С. 12–17.
71. Березный Е. А., Рубин А. М., Утехина Г. А. Практическая кардиоритмография. СПб. : НПО Нео, 2005. 140 с.
72. Malik M. Heart rate variability: Standards of measurement, physiological interpretation, and clinical use // Circulation. 1996. Vol. 93. Heart rate variability. № 5. P. 1043–1065.
73. Хаяутин В. М., Лукошкова Е. В. Спектральный анализ колебаний ЧСС – известное, спорное, неизвестное // Сб. научн. трудов I симпозиума. Миасс, 2000. С. 71–80.
74. Resting high-frequency heart rate variability is related to resting brain perfusion / B. Allen [et al.] // Psychophysiology. 2015. Vol. 52. № 2. P. 277–287.

75. Malpas S. C. Neural influences on cardiovascular variability: possibilities and pitfalls // *American Journal of Physiology – Heart and Circulatory Physiology*. 2002. Vol. 282. № 1. P. H6–H20.
76. Resting heart rate variability predicts self-reported difficulties in emotion regulation: a focus on different facets of emotion regulation / D. P. Williams [et al.] // *Frontiers in psychology*. 2015. Vol. 6. P. 261.
77. Флейшман А. Н. Вариабельность ритма сердца и медленные колебания гемодинамики. Нелинейные феномены в клинической практике. Новокузнецк, 2009. 262 с.
78. The forgotten role of central volume in low frequency oscillations of heart rate variability / M. Ferrario [et al.] // *PloS one*. 2015. Vol. 10. № 3. P. 0120167.
79. Julien C. The enigma of Mayer waves: Facts and models // *Cardiovascular Research*. 2006. Vol. 70. The enigma of Mayer waves. № 1. P. 12–21.
80. Unique very low-frequency heart rate variability during deep sleep in humans / F. Togo [et al.] // *IEEE Transactions on Biomedical Engineering*. 2006. Vol. 53. № 1. P. 28–34.
81. Usui H., Nishida Y. Relationship between physical activity and the very low-frequency component of heart rate variability after stroke // *Journal of Stroke and Cerebrovascular Diseases*. 2015. Vol. 24. № 4. P. 840–843.
82. Heart rate variability and treatment outcome in major depression: A pilot study / F. A. Jain [et al.] // *International Journal of Psychophysiology*. 2014. Vol. 93. Heart rate variability and treatment outcome in major depression. № 2. P. 204–210.
83. Хаспекова Н. Б. Диагностическая информативность мониторинга вариабельности ритма сердца // *Вестник аритмологии*. 2003. Т. 32. С. 15.
84. Смоленцев Н. К. Вейвлет-анализ в MATLAB. Т. 448. М., 2010. 628 с.
85. Egorova D. D., Kazakov Y. E., Kublanov V. S. Principal Components Method for Heart Rate Variability Analysis // *Biomedical Engineering*. 2014. Vol. 48. № 1. P. 37–41.
86. Richman J. S., Moorman J. R. Physiological time-series analysis using approximate entropy and sample entropy // *American Journal of Physiology-Heart and Circulatory Physiology*. 2000. Vol. 278. № 6. P. H2039–H2049.

87. Рюэль Д., Такенс Ф. О природе турбулентности // Странные аттракторы. М. : Мир, 1981. С. 117–151.
88. Ту Д. Принципы распознавания образов / под ред. Ю. И. Журавлева. М. : Мир, 1978. 411 с.
89. Дуда Р. Распознавание образов и анализ сцен: пер. с англ. М. : Мир, 1976. 511 с.
90. Вапник В. Н., Червоненкис А. Я. Теория распознавания образов: статистические проблемы обучения. М.: Наука., 1974.
91. Нильсон Н. Обучающиеся машины. М. : Мир, 1967. 180 с.
92. Фу К. С. Структурные методы в распознавании образов. М. : Мир, 1977. 320 с.
93. Гублер Е. В. Вычислительные методы анализа и распознавания патологических процессов. Л. : Медицина, 1978. 290 с.
94. Генкин А. А., Медведев В. И. Прогнозирование психофизиологических состояний. Вопросы методологии и алгоритмизации. Л. : Наука, 1973. 144 с.
95. Немирко А. П. Цифровая обработка биологических сигналов. М. : Наука, 1984. 144 с.
96. Айвазян С. А., Вежаева З. И., Староверов О. В. Классификация многомерных наблюдений. М. : Статистика, 1974. 240 с.
97. Васильев В. И. Распознающие системы. Киев : Наукова думка, 1983. 422 с.
98. Гонсалес Р., Вудс Р. Цифровая обработка изображений. Изд. 3-е, испр. и доп. М. : Техносфера, 2012. 1104 с.
99. Otsu N. A threshold selection method from gray-level histograms // IEEE transactions on systems, man, and cybernetics. 1979. Vol. 9. № 1. P. 62–66.
100. Roberts L. G. Machine Perception of three-dimensional solids // MIT Press, Cambridge, Mass, Optical and Electro-Optical Information Processing, 1965. URL: <http://www.packet.cc/files/mach-per-3D-solids.htm> Vol. 4 № 3. (дата обращения: 21.09.2019).
101. Prewitt J. M. Object enhancement and extraction // Picture processing and Psychopictorics. 1970. Vol. 10. № 1. P. 15–19.
102. Sobel I. E. Camera Models and Machine Perception. Ph. D. thesis. Palo Alto, California : Stanford University, 1970.
103. Marr D., Hildreth E. Theory of edge detection // Proceedings of the Royal Society of London. Series B. Biological Sciences. 1980. Vol. 207. № 1167. P. 187–217.

104. Canny J. A computational approach to edge detection // IEEE Transactions on pattern analysis and machine intelligence. 1986. № 6. P. 679–698.
105. Немирко А. П., Манило Л. А., Калиниченко А. Н. Математический анализ биомедицинских сигналов и данных. М. : Физматлит, 2017. 248 с.
106. Тихонов А. Н., Арсенин В. Я. Методы решения некорректных задач. М. : Наука, 1986. 288 с.
107. Hunt B. R. The application of constrained least squares estimation to image restoration by digital computer // IEEE Transactions on Computers. 1973. Vol. 100. № 9. P. 805–812.
108. Kaehler A., Bradski G. Learning OpenCV 3: computer vision in C++ with the OpenCV library. USA: O'Reilly Media, Inc., 2016. 990 p.
109. Lowe D. G. Distinctive image features from scale-invariant keypoints // International journal of computer vision. 2004. Vol. 60. № 2. P. 91–110.
110. Lindeberg T. Scale Invariant Feature Transform // Scholarpedia. 2012. Vol. 7. № 5. P. 10491.
111. Speeded-up robust features (SURF) / H. Bay [et al.] // Computer vision and image understanding. 2008. Vol. 110. № 3. P. 346–359.
112. Introduction to SURF (Speeded-Up Robust Features) – OpenCV 3.0.0-dev documentation. URL: https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_feature2d/py_surf_intro/py_surf_intro.html?highlight=surf (дата обращения: 05.11.2019).
113. Harris C. G., Stephens M. A combined corner and edge detector // Alvey vision conference. Citeseer. 1988. Vol. 15. P. 10–5244.
114. Brief: Binary robust independent elementary features / M. Calonder [et al.] // European conference on computer vision. Springer. 2010. P. 778–792.
115. Alcantarilla P. F., Bartoli A., Davison A. J. KAZE features // European Conference on Computer Vision. Springer, 2012. P. 214–227.
116. Alcantarilla P. F., Solutions T. Fast explicit diffusion for accelerated features in nonlinear scale spaces // IEEE Trans. Patt. Anal. Mach. Intell. 2011. Vol. 34. № 7. P. 1281–1298.
117. Weickert J., Scharr H. A scheme for coherence-enhancing diffusion filtering with optimized rotation invariance // Journal of Visual Communication and Image Representation. 2002. Vol. 13. № 1–2. P. 103–118.

118. Fleet D., Weiss Y. Optical flow estimation // Handbook of mathematical models in computer vision. Springer. 2006. P. 237–257.
119. Гантмахер Ф. Р. Теория матриц. М. : Наука, 1966. 576 с.
120. Форсайт Д., Понс Ж. Компьютерное зрение. Современный подход. М. : ИД Вильямс, 2004. 466 с.
121. Verkruysse W., Svaasand L. O., Nelson J. S. Remote plethysmographic imaging using ambient light // Optics express. 2008. Vol. 16. № 26. P. 21434–21445.
122. Viola P., Jones M. J. Robust real-time face detection // International journal of computer vision. 2004. Vol. 57. № 2. P. 137–154.
123. Pulse Rate Variability Analysis for Discrimination of Sleep-Apnea-Related Decreases in the Amplitude Fluctuations of Pulse Photoplethysmographic Signal in Children / J. Lázaro [et al.] // IEEE Journal of Biomedical and Health Informatics. 2014. Vol. 18. № 1. P. 240–246.
124. Кубланов В. С., Пуртов К. С. Дистанционная фотоплетизмография в задаче исследования variability сердечного ритма // Биомедицинская радиоэлектроника. 2015. № 8. С. 3–9.

Учебное издание

Кубланов Владимир Семенович
Долганов Антон Юрьевич
Костоусов Виктор Борисович
Немирко Анатолий Павлович
Манило Людмила Алексеевна
Петренко Тимур Сергеевич
Gamboa Hugo
Rodrigues João

**БИОМЕДИЦИНСКИЕ СИГНАЛЫ И ИЗОБРАЖЕНИЯ
В ЦИФРОВОМ ЗДРАВООХРАНЕНИИ:
ХРАНЕНИЕ, ОБРАБОТКА И АНАЛИЗ**

Редактор О. В. Климова
Верстка О. П. Игнатьевой

Подписано в печать 27.03.2020. Формат 70×100/16.
Бумага офсетная. Цифровая печать. Усл. печ. л. 19,4.
Уч.-изд. л. 12,1. Тираж 30 экз. Заказ 93.

Издательство Уральского университета
Редакционно-издательский отдел ИПЦ УрФУ
620049, Екатеринбург, ул. С. Ковалевской, 5
Тел.: +7 (343) 375-48-25, 375-46-85, 374-19-41
E-mail: rio@urfu.ru

Отпечатано в Издательско-полиграфическом центре УрФУ
620083, Екатеринбург, ул. Тургенева, 4
Тел.: +7 (343) 358-93-06, 350-58-20, 350-90-13
Факс: +7 (343) 358-93-06
<http://print.urfu.ru>



КУБЛАНОВ ВЛАДИМИР СЕМЕНОВИЧ, доктор технических наук, профессор УрФУ. Научные интересы: экспериментальная радиофизика, СВЧ-радиометры, обработка биомедицинских сигналов и изображений, нейроэлектростимуляторы в задачах реабилитации, математическое моделирование

ДОЛГАНОВ АНТОН ЮРЬЕВИЧ, кандидат технических наук, доцент УрФУ. Заместитель председателя IEEE R8 Russia Siberia. Научные интересы: машинное обучение, обработка биомедицинских сигналов

КОСТОУСОВ ВИКТОР БОРИСОВИЧ, кандидат физико-математических наук, заведующий отделом прикладных проблем управления ИММ УрО РАН. Научные интересы: навигация движущихся объектов по изображениям геофизических полей, обработка и распознавание изображений и сигналов

ПЕТРЕНКО ТИМУР СЕРГЕЕВИЧ, кандидат медицинских наук. Научные интересы: применение методов нейроэлектростимуляции в терапии психоневрологических расстройств у детей и взрослых

МАНИЛО ЛЮДМИЛА АЛЕКСЕЕВНА, доктор технических наук, профессор СПбГЭТУ «ЛЭТИ». Научные интересы: обработка биомедицинских сигналов, интеллектуальные биомедицинские системы

НЕМИРКО АНАТОЛИЙ ПАВЛОВИЧ, доктор технических наук, профессор СПбГЭТУ «ЛЭТИ». Научные интересы: распознавание образов, обработка и анализ биомедицинских сигналов

JOÃO RODRIGUES, PhD student at the Universidade Nova de Lisboa. Scientific interests: pattern search, analysis and generation in time series

HUGO GAMBOA, PhD, Assistant Professor at the Universidade Nova de Lisboa. Senior Scientist at Fraunhofer Portugal. Scientific interests: microelectronics, biosignal processing and software development