

Федеральное государственное автономное образовательное учреждение
высшего образования «ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ «ЮФУ»

На правах рукописи



Михайличенко Алексей Андреевич

**Методы и алгоритмы автоматизированной диагностики
остеоартрита по рентгенографическим изображениям**

Специальность 2.3.5 —

«Математическое и программное обеспечение вычислительных систем,
комплексов и компьютерных сетей»

Диссертация на соискание учёной степени
кандидата технических наук

Научный руководитель:

к.т.н., доцент

Демяненко Яна Михайловна

Ростов-на-Дону — 2023

Оглавление

	Стр.
Введение	6
Глава 1. Данные и методы	15
1.1 Существующие подходы к диагностике остеоартрита коленного сустава	15
1.2 Обзор наборов данных для исследования остеоартрита коленного сустава	19
1.2.1 Датасет COCO	20
1.2.2 Датасет OAI	20
1.2.3 Датасет MOST	22
1.2.4 Датасет CHECK	23
1.3 Обзор современных подходов и идей в проектировании сверточных нейронных сетей	25
1.3.1 AlexNet	26
1.3.2 ZFNet, VGG	28
1.3.3 ResNet	29
1.3.4 Inception	32
1.3.5 Inception-ResNet	36
1.3.6 Xception	38
1.3.7 ResNeXt	41
1.3.8 DenseNet	42
1.3.9 Squeeze-and-Excitation Networks	45
1.4 Обзор методов оценки точности локализации и классификации . .	52
1.4.1 Оценка качества локализации	52
1.4.2 Оценка качества классификации	53
1.5 Обзор методов автоматической локализации коленного сустава . .	61
1.6 Обзор методов автоматической классификации остеоартрита коленного сустава	68
1.7 Обзор методов выделения контуров на изображении	80
1.8 Выводы	84

Глава 2. Разработка и исследование метода автоматической локализации коленного сустава	86
2.1 Описание метода	87
2.1.1 Архитектура Single Shot Detector	87
2.1.2 Архитектура MobileNet	90
2.2 Эксперимент	93
2.3 Выводы	96

Глава 3. Разработка и исследование метода автоматической классификации остеоартрита	97
3.1 Схема и параметры обучения нейронных сетей	97
3.1.1 Функция потерь	98
3.1.2 Аугментация	100
3.2 Разработка метода автоматической классификации остеоартрита коленного сустава	101
3.2.1 Классификация остеоартрита методом WND-CHARM	102
3.2.2 Классификация при помощи простой сверточной нейронной сети	104
3.2.3 Использование классических архитектур сверточных нейронных сетей и новых архитектурных решений	108
3.3 Использование ансамблей для улучшения точности классификации остеоартрита коленного сустава	117
3.3.1 Обзор существующих подходов агрегации результатов работы нескольких классификаторов	119
3.3.2 Эксперимент	123
3.4 Обсуждение результатов	128
3.5 Выводы	134

Глава 4. Разработка метода детектирования контуров костей на медицинских рентгенографических изображениях	135
4.1 Разработка метода выделения контуров костей на рентгенограмме	135
4.1.1 Сглаживание и фильтрация шума	136
4.1.2 Вычисление вспомогательной маски объектов	137

4.1.3	Вычисление градиента изображения и поиск «сильных» фрагментов границы объектов	138
4.1.4	Устранение разрывов и объединение фрагментов границ в контуры	141
4.1.5	Метод отслеживания контура на изображении	147
4.1.6	Уточнение и сглаживание финальных контуров	149
4.1.7	Количественная оценка качества выделения контуров	152
4.1.8	Поиск оптимального порога бинаризации	153
4.2	Эксперименты	154
4.3	Выводы	156
Заключение		158
Список литературы		160
Список рисунков		176
Список таблиц		183
Словарь терминов		187
Приложение А. Структура сверточных нейронных сетей различных архитектур		188
Приложение Б. Таблицы со значениями метрик классификации для традиционных архитектур сверточных нейронных сетей		193
Приложение В. Свидетельство о государственной регистрации программы для ЭВМ №2022665581		205
Приложение Г. Акты о внедрении результатов работы		207
Приложение Д. Описание и краткая инструкция к разработанному программному обеспечению		214
Д.1	Общее описание	214
Д.2	Базовый функционал	215

	Стр.
Д.2.1 Проведение измерений	216
Д.2.2 Редактирование изображения	219
Д.3 Расширенный функционал	219
Д.3.1 Особенности работы с модулем локализации	220
Д.3.2 Выделение контуров	220
Д.3.3 Особенности работы с модулями классификации	221
Д.4 Общее описание ПО	221

Введение

Актуальность темы исследования. Цифровая обработка изображений и компьютерное зрение сегодня являются важными разделами прикладной математики и информатики. Методы и алгоритмы цифровой обработки изображений находят широкое применение в самых разнообразных областях: в системах контроля качества, робототехнике, системах автономного вождения, медицинской диагностике и других.

Обработка медицинских изображений, получаемых при помощи компьютерной и магнитно-резонансной томографии, рентгенографии и ультразвуковых исследований является одним из самых популярных современных направлений исследований в области цифровой обработки изображений.

В настоящее время в России для медицинской диагностики широко используются рентгенографические изображения, при этом практически вся обработка проводится вручную. Это требует высокой квалификации специалистов, а в случае массового потока изображений приводит к сильной загруженности рентгенологических кабинетов, что в свою очередь повышает вероятность ошибок диагностики из-за человеческого фактора. Использование методов автоматизированной диагностики позволяет избавиться от фактора субъективности и сократить количество изображений, требующих внимания специалиста.

Существующие методы обработки изображений для компьютерной диагностики остеоартрита (ОА) можно разделить на две группы.

Методы из первой группы обычно основаны на использовании комбинации классических алгоритмов обработки и анализа изображений (размытие, анализ текстурных характеристик, выделение контуров и др.), и позволяют получить некоторые количественные оценки рассматриваемых критериев (расстояние между объектами, углы, особенности расположения ключевых точек и т.п.). Подобные методы активно развивались научной школой В.С. Пилиди, а также подробно описаны в работах Х. Ока (H. Oka), Ш. Мураки (S. Muraki) и Т. Акунэ (T. Akune).

Вторая группа методов опирается на алгоритмы машинного обучения. Данные методы более универсальны и в меньшей степени зависят от качества исходных данных, источника этих данных и прочих варьирующихся характеристик, что представляет отдельную проблему для методов первой группы.

В современном виде для решения задачи автоматической диагностики ОА они были предложены в работах Д. Энтони (J. Antony) и К. МакГиннеса (K. McGuinness), а затем получили развитие в исследованиях А. Тюльпина.

К недостаткам методов автоматизированной диагностики, опирающихся на классические алгоритмы обработки изображений, можно отнести некорректные результаты их работы в случае плохого качества данных и при нестандартной форме объектов, которая возникает из-за различных заболеваний. Классические методы машинного обучения, из-за своей специфики, обладают высокими требованиями к вычислительным ресурсам, что ограничивает их применение в медицинских учреждениях с недостаточно мощным оборудованием.

Поэтому, **актуальной** задачей является создание новых методов и алгоритмов обработки цифровых медицинских рентгенографических изображений для автоматизированной диагностики заболеваний, обладающих высокой точностью работы и способных работать в условиях ограниченных вычислительных ресурсов.

Объектом исследования являются системы автоматизированной диагностики остеоартрита по рентгенографическим изображениям.

Предметом исследования являются методы и алгоритмы компьютерного зрения, цифровой обработки изображений и глубокого обучения для автоматизированной диагностики остеоартрита.

Целью диссертации является повышение точности автоматизированной диагностики остеоартрита по цифровым рентгенограммам.

Общая научная задача: разработка методов и алгоритмов повышения точности автоматизированной диагностики остеоартрита по цифровым рентгенограммам без использования априорной информации о форме объектов в условиях ограниченных вычислительных ресурсов.

Для достижения поставленной цели необходимо решить следующие **задачи**.

1. Провести анализ существующих методов обработки изображений для автоматизированной диагностики остеоартрита и алгоритмов выделения контуров объектов на медицинских рентгенограммах. Исследовать применимость известных архитектур сверточных нейронных сетей для решения задач автоматической локализации сустава и диагностики ОА на медицинских рентгенограммах, а также провести вычислительные эксперименты по оценке их эффективности.

2. Разработать метод автоматической локализации сустава на рентгенограмме, который обладает повышенной точностью в условиях ограниченных вычислительных ресурсов.
3. Разработать метод повышения точности автоматической диагностики остеоартрита сверточными нейронными сетями по цифровым медицинским изображениям.
4. Разработать новый алгоритм детектирования контуров костей на медицинских рентгенограммах, не требующий априорной информации о форме объектов.
5. Разработать программное обеспечение для реализации предложенных методов и алгоритмов, и провести анализ их эффективности.

Методология и методы исследования. В исследовании использовались методы теории вероятностей и компьютерного зрения, методы оптимизации, теория распознавания образов, машинное и глубокое обучение, статистические методы оценки результатов экспериментов и другие.

Научная новизна. При решении поставленных задач в рамках исследовательской работы были получены следующие **новые научные результаты**:

1. предложен метод локализации коленного сустава на рентгенограмме на базе нейросетевой архитектуры Single Shot Detector (SSD), **отличающийся** от существующих методов локализации более высокой точностью работы и более низкими требованиями к вычислительным ресурсам;
2. разработан метод повышения точности автоматической диагностики остеоартрита нейросетями, использующий комплекс архитектурных решений (интеграция блоков сжатия и возбуждения, использование ансамблей, модификация шкалы оценки), **отличающийся** от существующих методов более высокой точностью и стабильностью работы;
3. разработан новый алгоритм выделения контуров костей на медицинских рентгенографических изображениях, **отличающийся** от существующих тем, что не использует информацию об априорной форме объекта, что позволяет работать с сильно искаженными объектами и нестандартными конфигурациями;
4. предложены **новый** алгоритм отслеживания контуров на бинарном изображении, **отличающийся** от существующих большей устойчивостью к артефактам, и **новая** энергия для метода активных контуров,

учитывающая направление градиента при изменении положения точек, что позволяет более точно проводить уточнение контура.

Основные положения, выносимые на защиту:

1. разработанный метод локализации сустава на базе нейросетевой архитектуры Single Shot Detector (SSD) позволяет в условиях ограниченных вычислительных ресурсов ускорить обработку и повысить точность классификации остеоартрита;
2. разработанный метод повышения точности автоматической классификации остеоартрита, использующий комплекс архитектурных решений (интеграция блоков сжатия и возбуждения в структуру нейросети, использование ансамблей, модификация шкалы оценки), позволяет сократить количество рентгенограмм, требующих внимания специалиста;
3. разработанный алгоритм выделения контуров костей на рентгенограмме, не использующий априорную информацию о форме объекта, может работать с сильно искаженными объектами и позволяет повысить точность необходимых для диагностики измерений.

Теоретическая значимость. Результаты, полученные в диссертации, являются важными для развития методов автоматизированной диагностики остеоартрита по рентгенографическим изображениям.

Автором **показано**, что разработанный метод повышения точности автоматической классификации ОА, использующий предложенный комплекс архитектурных решений, может быть использован для улучшения качества алгоритмов классификации ОА, основанных на использовании сверточных нейронных сетей.

Автором **показано**, что разработанный алгоритм детектирования контуров, использующий новый метод отслеживания контуров и предложенную энергию для активных контуров, не требует априорной информации о форме объектов и может быть использован для работы с сильно искаженными объектами.

Практическая значимость. Разработанные методы, алгоритмы и их программная реализация могут быть использованы для анализа цифровых рентгенографических изображений как в интерактивном, так и в автоматическом режимах в условиях большой загруженности, показывая высокую точность

диагностики и существенно уменьшая количество рентгенограмм, требующих внимания специалиста.

1. Предложенный метод локализации сустава на базе SSD может работать в условиях ограниченных вычислительных ресурсов и на тестовой выборке из датасета OAI для коэффициента перекрытия искомого и найденного прямоугольников не менее 0,75 показал точность 94,1%, что на 1,9% превышает точность метода, основанного на использовании модели YOLOv2, на 4,9% — точность метода на базе полностью сверточных нейронных сетей и более чем в 11 раз превышает точность методов, основанных на сопоставлении с шаблоном.
2. Реализованный метод повышения точности классификации на основе комбинации архитектурных решений позволил существенно повысить точность классификации остеоартрита сверточными нейронными сетями: изменение шкалы оценки позволило повысить точность от 9,86% до 14,39% при сохранении информативности диагностики, интеграция блоков сжатия и возбуждения дала прирост до 3,85%, а использование ансамблей позволило повысить точность на более чем 2,4%. Финальная точность классификации составила 72,28% для случая 5 классов для ансамбля из трех SE-ResNeXt-50, 84,66% для 4 классов для ансамбля из трех SE-DenseNet-121 и 87,5% для бинарной классификации для ансамбля из трех SE-Xception, что превосходит существующие методы классификации для всех рассматриваемых случаев.
3. Разработанный алгоритм выделения контуров корректно выделил контуры для 84,96% объектов, и еще для 5,08% результат признан пригодным к использованию после небольшой корректировки спорных участков экспертом. Результаты работы алгоритма могут быть использованы для автоматизации измерений, необходимых для получения количественных оценок ОА: измерения сужения межсуставной щели, определения «параллельности» кромки сустава и др.
4. Созданный на базе предложенных методов программный комплекс может быть использован как для проведения стандартных медицинских измерений, так и для выполнения автоматической диагностики остеоартрита путем классификации по разным шкалам оценки (в том числе в режиме пакетной обработки) с целью обнаружения болезни на ранних стадиях. Использование автоматических методов анализа позволяет

облегчить работу эксперта, устранить фактор субъективности и компенсировать недостаток опыта при принятии решения.

Помимо классификации, предложенные модели сверточных нейронных сетей могут быть использованы для получения распределения вероятностей классов ОА и акцентирования внимания специалиста на определенных областях рентгенограммы [144].

Достоверность полученных результатов обеспечивается проведением серий экспериментов для каждой модели нейронной сети на разных наборах данных. В каждом из экспериментов исследуемые модели обучались с разным начальным значением генератора случайных чисел и на разных комбинациях подмножеств тренировочной и валидационной выборок, что позволяет обучать разные сети, не меняя при этом их структуры. Устойчивость результатов классификации обеспечивается усреднением результатов работы нескольких моделей. Кроме того, результаты находятся в соответствии с результатами, полученными другими авторами в данной области исследований.

Реализация и внедрение результатов работы. Результаты диссертации были внедрены в рабочие процессы ООО «Наука», акт от 2 декабря 2022, утвержден заместителем генерального директора ООО «Наука». Разработанное в ходе написания диссертации программное обеспечение было адаптировано под решение специальных задач и использовано при проведении медицинских исследований.

Также, полученные результаты исследования были внедрены при выполнении ряда работ в ООО «Изоскан», акт от 5 октября 2022 года, утвержден техническим директором ООО «Изоскан». Результаты диссертации были использованы при реализации программного комплекса, решающего задачу автоматической классификации объектов по графическим данным. Внедрение результатов исследования позволило реализовать эффективный алгоритм классификации, точность работы которого превышает 90%.

Кроме того, результаты диссертационного исследования были внедрены в учебном процессе кафедры прикладной математики и программирования (ПМП) Института математики, механики и компьютерных наук (ИММиКН) Южного федерального университета (ЮФУ), акт от 19 сентября 2022 года, утвержден директором ИММиКН. Материалы диссертации использовались в лекционном курсе по дисциплине «Компьютерное зрение», лекционном курсе по дисциплине «Машинное обучение» и лекционном курсе по дисциплине «Нейронные сети

для мобильных приложений» для подготовки магистров направления подготовки 02.04.02 «Фундаментальная информатика и информационные технологии».

Апробация работы. Основные результаты работы докладывались и обсуждались на следующих конференциях:

- Первая международная научная конференция «Осенние математические чтения в Адыгее», г. Майкоп, 2015 [145].
- XXIII международная конференция «Математика. Компьютеры. Образование», г. Дубна, 2015.
- The 5th International Conference on Image Analysis, Social Networks and Texts (AIST 2016), г. Екатеринбург, 2016 [140].
- 26-я Международная конференция по компьютерной графике и обработке изображений (ГрафиКон 2016), г. Нижний Новгород, 2016 [146].
- Всероссийская конференция «Современные информационные технологии: тенденции и перспективы развития», г. Ростов-на-Дону, 2016.
- 27-я Международная конференция по компьютерной графике и обработке изображений (ГрафиКон 2017), г. Пермь, 2017.
- XXVII научная конференция «Современные информационные технологии: тенденции и перспективы развития» (СИТО 2020), г. Ростов-на-Дону, 2020 [147].
- The 9th International Conference on Image Analysis, Social Networks and Texts (AIST), г. Москва, 2020 [141].
- XXVIII научная конференция «Современные информационные технологии: тенденции и перспективы развития» (СИТО 2021), г. Ростов-на-Дону, 2021 [144].
- XXIX научная конференция «Современные информационные технологии: тенденции и перспективы развития» (СИТО 2022), г. Ростов-на-Дону, 2022 [148].
- 30-я международная конференция «Математика. Компьютер. Образование» (МКО 2023), г. Дубна, 2023.

Соответствие паспорту специальности. Содержание диссертации соответствует пункту 7 («Модели, методы, архитектуры, алгоритмы, форматы, протоколы и программные средства человеко-машинных интерфейсов, компьютерной графики, визуализации, обработки изображений и видеоданных, систем виртуальной реальности, многомодального взаимодействия в социкиберфизических системах») в части «методы и алгоритмы обработки изображений», а

также пункту 4 («Интеллектуальные системы машинного обучения, управления базами данных и знаний, инструментальные средства разработки цифровых продуктов») в части «интеллектуальные системы машинного обучения» паспорта научной специальности 2.3.5 — «Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей».

Личный вклад. Автор самостоятельно провел теоретические изыскания и осуществил практическую реализацию всех описанных в диссертации алгоритмов и методов. Были проведены все необходимые эксперименты и выполнен анализ полученных результатов. По результатам работы разработан программный комплекс, пригодный для использования в медицинской практике.

Публикации. Основные результаты по теме работы изложены в 10 печатных изданиях, 1 из которых изданы в журналах, рекомендованных ВАК, 4 — в научных журналах и изданиях, индексируемых Scopus, 5 — в сборниках трудов конференций и тезисах докладов. Зарегистрирована 1 программа для ЭВМ.

Объем и структура работы. Диссертация состоит из введения, 4 глав и заключения. Полный объем диссертации составляет 222 страницы, включая 79 рисунков, 58 таблиц. Список литературы содержит 150 наименований.

В первой главе обсуждаются существующие архитектуры сверточных нейронных сетей и приводится анализ применимости той или иной архитектуры к решению задачи классификации остеоартрита в условиях ограниченного размера датасетов для обучения и тестирования. Для каждой из архитектур приводится подробное описание ее структуры и важные архитектурные особенности, отличающие ее от остальных архитектур.

Дано описание существующих наборов данных, используемых в исследовании, а также методов диагностики и методологий оценки стадии прогрессирования остеоартрита, включая шкалу Косинской и атлас международного общества по изучению остеоартрита (OARSI).

В конце главы приводится описание существующих методов локализации области коленного сустава на рентгенограмме и методов автоматической классификации остеоартрита. Для каждого метода указаны результаты, которых удалось достичь в исследованиях, а также его сильные и слабые стороны. Описаны используемые методы оценки качества классификаторов и методы оценки качества локализации коленного сустава на рентгенограмме.

Вторая глава посвящена задаче автоматической локализации области коленного сустава на рентгенографическом изображении. Предлагается новый

метод, основанный на использовании сверточной нейронной сети с архитектурой Single Shot Detector (SSD). Данная архитектура является сетью прямого распространения, которая на выходе возвращает фиксированный набор ограничивающих прямоугольников и оценку вероятности наличия в этих прямоугольниках искомого объекта. В рамках исследования проводится реализация и обучение SSD на датасете, содержащем рентгенограммы коленей, и выполняется сравнение результатов работы обученной сети на тестовой выборке с методами из ранее опубликованных работ при помощи коэффициента Жаккара (пересечение над объединением между искомым и найденным прямоугольниками).

В **третьей главе** описывается предложенный метод повышения точности автоматической классификации остеоартрита сверточными нейронными сетями при помощи комплекса архитектурных решений (использование функции потерь специального вида, модификация шкалы оценки ОА, интеграция блоков сжатия и возбуждения, использование ансамблей).

В **четвертой главе** предлагается новый метод выделения контуров костей коленного сустава на рентгенограмме, не требующий критерия однородности областей, который использует в своей работе комбинацию различных алгоритмов обработки изображений, таких как размытие, пороговое разделения, вычисление краев, метод активных контуров и др. Описываются новые алгоритмы устранения разрывов границы на изображении, алгоритм отслеживания контуров, предлагается дополнительная энергия в методе активных контуров для уточнения финальной конфигурации объектов и предлагается численный критерий оценки качества выделения контуров на изображении. Предложенные методы, алгоритмы и критерии оценки показывают лучший результат в исследуемой области по сравнению с существующими аналогами.

В **заключении** приведены основные результаты диссертационной работы.

Глава 1. Данные и методы

В данной главе описываются существующие методы автоматизированной диагностики остеоартрита и связанные с ней разделы.

Сначала приводится описание существующих подходов к оценке степени прогрессирования остеоартрита коленного сустава и существующих наборов размеченных данных для проведения исследований в этой области. Также, приводится обоснование для выбора датасета The Osteoarthritis Initiative (OAI) в качестве основного набора данных для проведения данного исследования.

Дан обзор основных современных архитектур сверточных сетей, которые далее будут использованы применительно к задаче автоматической классификации остеоартрита коленного сустава. Для каждой такой архитектуры приводится подробное описание их особенностей и отличий по сравнению с предыдущими архитектурами, а также представлено подробное описание структуры соответствующих моделей в виде таблицы или схемы.

Представлено описание существующих методологий оценки качества классификации [148; 139] и локализации, и рассказано об используемых в диссертации метриках.

В конце главы приводится подробное описание существующих методов локализации коленного сустава, автоматической диагностики остеоартрита и методов детектирования контуров костей на медицинских изображениях.

1.1 Существующие подходы к диагностике остеоартрита коленного сустава

Основные признаки остеоартрита — дегенерация и износ суставного хряща, вызванные замедлением в нем репаративных процессов, в результате различного рода изменений. Отсюда и основные части тела, которые ему подвержены — это коленный, тазобедренный и плечевой суставы, а также пальцы рук. Из-за того, что в хряще отсутствуют нервные окончания, изменения в суставе на первых стадиях заболевания протекают бессимптомно. Симптомы проявляются лишь при вовлечении в процесс тканей, содержащих нервы, или непосредственно кости. Однако хрящевые ткани не видны непосредственно

на рентгенограммах, а прогрессирование ОА оценивается в основном по косвенным признакам — сужению суставов или появлению остеофитов и других костных девиаций.

Методы диагностики остеоартрита можно разделить на количественные и комбинированные [1]. Самым распространенным количественным методом является измерение межсуставного пространства между контурами бедренной и большеберцовой костей. Раньше для этого использовали обыкновенную линейку, сейчас же, с широким распространением цифровизации в сфере медицины, существуют специальные программные средства, облегчающие этот процесс для исследователя (рис. 1). Стандартной мерой оценки степени прогрессирования остеоартрита является вычисление минимального значения межсуставного пространства, однако некоторые исследователи используют либо среднее значение, либо расстояние между контурами костей в определенных областях. Минимальное значение при этом более чувствительно к изменениям, чем среднее [2].

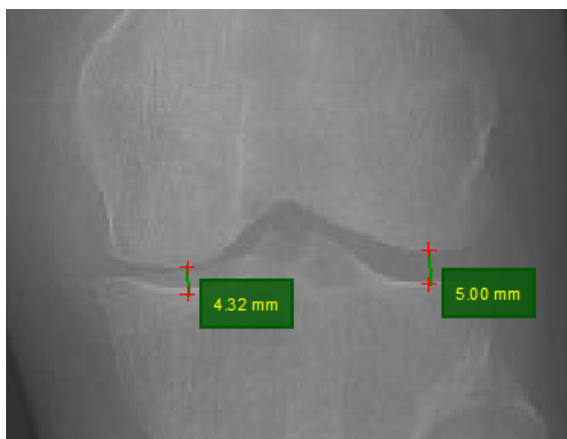


Рисунок 1 — Измерение сужения межсуставного пространства в специализированном программном обеспечении

Известно, что на сужение межсуставного пространства, помимо остеоартрита, также могут влиять разного рода повреждения мениска (например, вывихи) и прочие подобные заболевания. Кроме того, к одним из ограничений можно отнести разные вариации положения колена в полусогнутом состоянии. Эти факторы могут оказывать значительное влияние на измеренное межсуставное пространство и, соответственно — на оценку степени ОА.

Среди комбинированных методов к оценке развития ОА существуют различные подходы. В отечественном пространстве ранее широко использовалась клинико-рентгенологическая классификация Н.С. Косинской [3], в Европе и Северной Америке в основном используется шкала Келлгрена-Лоуренса (KL), в

некоторых случаях для определения прогрессирования болезни используется атлас международного общества изучения остеоартрита (OARSI) и др.

В отличие от шкалы Келлгрена-Лоуренса, классификация Косинской, которая состоит из трех стадий, учитывает клинические проявления и симптомы остеоартроза (табл. 1), однако размеченных датасетов, работающих с классификацией Косинской, не существует. Кроме того, составление подобного датасета осложнялось бы необходимостью собирать клинические проявления болезни, озвученные пациентами, что вносило бы определенную долю субъективизма в данные.

Таблица 1 — Классификация остеоартрита по Косинской

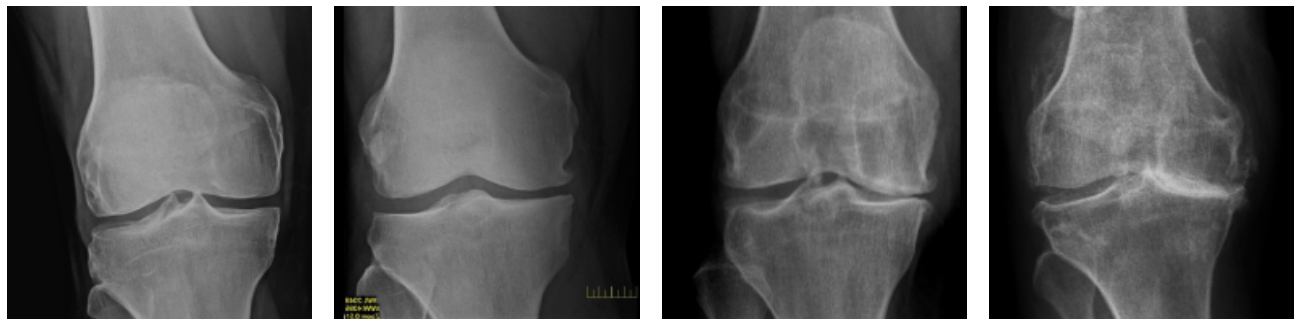
Стадия	Словесное описание
I стадия	На рентгенограмме определяется незначительное сужение суставной щели по сравнению со здоровым суставом и лёгкий субхондральный остеосклероз. Клинические проявления: боль, возникающая при ходьбе, особенно при подъёме или спуске по лестнице, которая проходит в состоянии покоя, движения в суставе, обычно, не ограничены.
II стадия	Сужение суставной щели в 2–3 раза превышает норму, субхондральный склероз становится более выражен, по краям суставной щели появляются костные наросты (остеофиты). Клинические проявления: гипотрофия мышц, хромота, умеренный болевой синдром, развивается ограничение движений в суставе, определяется лёгкая фронтальная деформация оси конечности.
III стадия	На рентгенограмме суставная щель почти полностью отсутствует, определяются обширные костные разрастания, наблюдается значительная деформация и склерозирование суставных поверхностей. Клинические проявления: резко выраженные боли и хромота, нестабильность сустава и атрофия мышц бедра и голени, невозможность полного сгибания или разгибания конечности в суставе.

Шкала Келлгрена-Лоуренса также находит широкое применение и в иностранной, и в отечественной медицинской практике, поэтому в данной работе исследование будет проводиться при использовании шкалы KL. Критика данной системы оценивания направлена на акцентирование внимания на остеофитах, общую градацию степеней ОА, а также нечувствительность к изменениям [4]. Также в некоторых работах приводятся доказательства того, что оценка Келлгрена-Лоуренса может значительно различаться между разными экспертами [5]. Несмотря на это, данная система является одним из «золотых стандартов» при оценке развития остеоартрита. В ней выделяется 5 стадий развития ОА, включая стадию 0 — отсутствие остеоартрита. Описание каждой из стадий приводится в таблице 2. Примеры рентгенограмм для каждой из степеней шкалы Келлгрена-Лоуренса представлены на рис. 2.

Таблица 2 — Стадии развития ОА по шкале Келлгрена-Лоуренса

Стадия	Словесное описание
Стадия 0	<i>none</i> — изменения отсутствуют
Стадия 1	<i>doubtful</i> — сомнительные рентгенологические признаки: сужения суставной щели нет или определяется небольшое сужение, заострения или небольшие остеофиты на краях суставной поверхности
Стадия 2	<i>minimal</i> — минимальные изменения: небольшое сужение суставной щели, небольшие единичные остеофиты на краях суставной поверхности
Стадия 3	<i>moderate</i> — умеренные проявления: умеренное сужение суставной щели, множественные остеофиты на краях суставной поверхности, небольшие деформации суставной поверхности
Стадия 4	<i>severe</i> — выраженные изменения: суставная щель почти не прослеживается, грубые остеофиты на краях суставной поверхности, разной степени выраженности деформации эпифизов костей, образующих сустав

Чтобы увеличить полезность рентгенограмм, а также с целью облегчения работы специалистов, были разработаны специальные атласы, которые



Стадия 1

Стадия 2

Стадия 3

Стадия 4

Рисунок 2 — Система Келлгрена-Лоуренса для оценки прогрессирования остеоартрита коленного сустава

могут быть использованы в качестве ориентиров при оценке индивидуальных особенностей ОА [4; 6—8]. Атлас международного общества по изучению остеоартрита (the Osteoarthritis Research Society International, OARSI) был опубликован в 1996 году. В 2007 данный атлас был обновлен новыми изображениями, полученными на современном оборудовании и имеющими лучшее качество [4].

Наличие у пациента остеоартрита диагностируют при стадии 2 и выше [9], поэтому во многих исследованиях стадии 0 и 1 часто объединяют в одну, означающую отсутствие остеоартрита. Несмотря на то, что в этом случае шкала Келлгрена-Лоуренса и шкала, основанной на атласах OARSI, имеют одинаковое количество стадий развития ОА (0 — doubtful, 1 — minimal, 2 — moderate, 3 — severe), нельзя считать их эквивалентными. В работе [10] приводятся аргументы о некорректности точного сопоставления подобных шкал оценок, а в заключении сообщается, что данная гипотеза нуждается в более широком рассмотрении и проверке при помощи МРТ.

1.2 Обзор наборов данных для исследования остеоартрита коленного сустава

Несмотря на то, что остеоартрит является одним из самых распространенных заболеваний опорно-двигательного аппарата, какие-либо мероприятия по сохранению и систематизации клинических, рентгенологических и т.п. данных с целью их дальнейшего использования начали осуществляться относительно недавно. В частности, для исследований остеоартрита в области машинного обу-

чения де-факто стандартом являются наборы данных The Osteoarthritis Initiative (OAI) и Multicenter Osteoarthritis Study (MOST), которые используются в подавляющем большинстве работ. Также, относительно недавно в открытом доступе появился датасет Cohort Hip & Cohort Knee (CHECK), который содержит в себе как клинические данные нескольких тысяч пациентов, так и наборы рентгенограмм коленей в разных проекциях, бедер, кистей рук и т.п.

1.2.1 Датасет COCO

Датасет COCO (Common Objects in Context) — набор данных, созданный и поддерживаемый корпорацией Microsoft, который широко используется для решения задач таких задач компьютерного зрения, как сегментация, локализация, классификация и других [11]. Датасет включает в себя более 330К изображений, из которых размечено более 200К. Изображения содержат в себе информацию о более чем полутора миллионах объектов, разделенных на 80+ категорий, с различными метками к каждому изображению. В работе датасет COCO использовался опосредованно — использовались предварительно обученные на этом датасете модели нейронных сетей.

1.2.2 Датасет OAI

Датасет OAI (The Osteoarthritis Initiative), созданный при финансовой поддержке национального института здоровья США (the National Institutes of Health), является одним из основных датасетов для проведения исследований в области изучения остеоартрита по рентгеновским снимкам коленного сустава. Он содержит архив клинических данных 4492 пациентов, мужчин и женщин, в возрасте от 45 до 79 лет, наблюдаемых в рамках программы в течение 10 лет (табл. 3).

Данные включают в себя рентгенограммы (рентгеновские снимки), магнитно-резонансные изображения (МРТ), а также клиническую информацию о пациентах (пол, возраст, рост и т.п.) и различные количественные оценки ОА для

Таблица 3 — Распределение изображений датасета OAI по возрасту и полу

Группа	45–49 лет	50–59 лет	60–69 лет	70–79 лет	Всего
Мужчины	240	705	476	461	1882
Женщины	270	861	893	586	2610
Всего	510	1566	1369	1047	4492

каждого изображения — ширину межсуставного пространства (joint space width, JSW) в шкале OARSI для средней и боковой части сустава, средний минимум JSW в миллиметрах, стадия OA по шкале Келлгрена-Лоуренса и др. Для каждой рентгенограммы, в ходе двухэтапного процесса оценки (первичным и вторичным экспертами), была определена степень OA по шкале Келлгрена-Лоуренса и степень сужения межсуставного пространства по шкале OARSI. Для исследования в рамках данной работы использовался набор рентгенограмм, содержащих двухстороннюю задне-переднюю проекцию коленей с фиксированным сгибанием (bilateral posterior-anterior (PA) fixed-flexion knees), полученные на этапе предварительной оценки состояния пациентов. Примеры рентгенограмм представлены на рис. 3.



Рисунок 3 — Примеры полных рентгенограмм коленей в двухсторонней задне-передней проекции с фиксированным сгибанием

Несмотря на обширный набор разнообразной информации о рентгенограммах, которые имеются в датасете, в нем отсутствует информация о положении межсуставных щелей между бедренной и большеберцовой костями, которая необходима для дальнейших исследований автоматическими методами. Автоматический анализ можно выполнять и по полной рентгенограмме, однако в этом случае точность может получиться ниже из-за наличия на изображении лишней информации, которая напрямую о признаках остеоартрита ничего не сообщает, но усложняет анализ.

Для того, чтобы дополнить датасет недостающей информацией, **в рамках диссертации** была проведена ручная разметка всех 4796 рентгенограмм, где для каждой рентгенограммы был сформирован набор из одного или двух ограничивающих прямоугольников (в зависимости от количество суставов на изображении), которые указывают точное положение межсуставных щелей (рис. 4). Это позволило получить 9557 различных примеров суставов из всего набора рентгенограмм.

Для удобства дальнейшего использования, полученные прямоугольники затем были расширены по высоте, чтобы сделать ее равной ширине (большинство используемых в работе архитектур сверточных нейронных сетей принимают на вход изображения с одинаковыми размерностями по ширине и высоте).

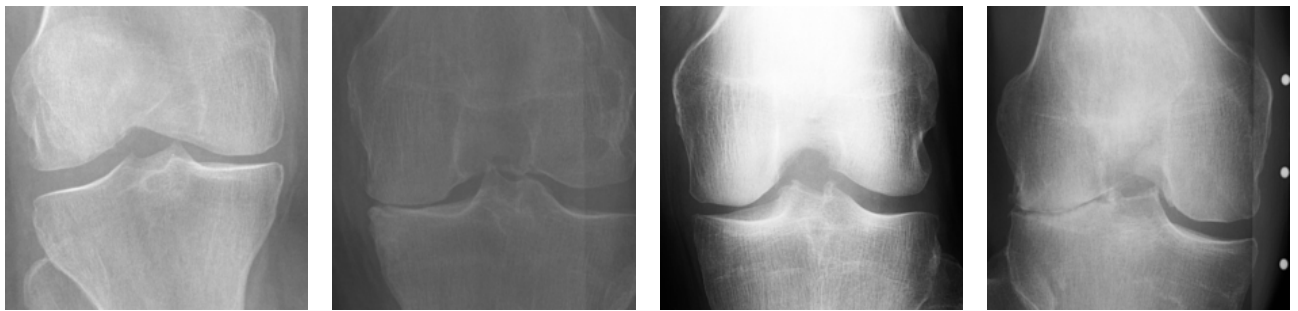


Рисунок 4 — Примеры изображений области коленного сустава, полученных в результате разметки из исходных рентгенограмм

После ручной фильтрации данных, в результате которой из рассмотрения были отброшены изображения плохого качества (имеющие сильные засветы, размытость и другие дефекты), был сформирован новый датасет из 8260 отдельных изображений сустава. Затем этот датасет был разбит на тренировочную, валидационную и тестовую выборки в соотношении 7 : 1 : 2 соответственно, точный состав которой указан в таблице 4.

1.2.3 Датасет MOST

Multicenter Osteoarthritis Study — проспективное исследование остеоартрита коленного сустава, спонсируемое Национальным институтом здоровья США,

Таблица 4 — Распределение изображений, использовавшихся для экспериментов. Числа в таблице обозначают число изображений суставов, имеющих в каждой из групп

Выборка	KL-0	KL-1	KL-2	KL-3	KL-4	Всего
Тренировочная	2295	1051	1505	752	175	5778
Валидационная	319	148	223	111	25	826
Тестовая	639	296	447	223	51	1656
Всего	39,38%	18,11%	26,32%	13,15%	3,04%	100%

у пожилых людей с этой болезнью или повышенными рисками ее возникновения, целью которого является изучение возможностей ранней диагностики и лечения заболевания [12].

Исследование включает в себя данные о 3026 пациентах в возрасте от 50 до 79 лет (из которых около 60% — женщины), наблюдение за которыми велось в течение 84 месяцев. Помимо базового исследования проводились дополнительные обследования через 15, 30, 60 и 84 месяца, включающие в себя оценку факторов риска, характеристик текущей стадии остеоартрита, разного рода измерения, а также радиологические данные (рентгенограммы и MPT). Кроме того, дополнительно проводилось телефонное интервью пациентов на 72-м месяце.

К сожалению, публичный доступ к набору данных MOST на момент написания диссертации был прекращен в связи с реорганизацией и прекращением финансирования отделения, занимавшегося его сопровождением, поэтому использовать его для расширения базового набора OAI (что сделано в некоторых исследованиях) больше не представляется возможным, и поэтому эти данные в работе не используются.

1.2.4 Датасет CHECK

Датасет CHECK (Cohort Hip & Cohort Knee) является результатом многолетнего наблюдения [13] за группой пациентов возрастом 45–65 лет, с ранними симптомами остеоартрита коленей и бедер в Нидерландах при финансовой поддержке The Dutch Arthritis Foundation. Наблюдение за пациентами осуществлялось в течение 10 лет, исследования содержат как различные клинические

данные, так и радиографические, и биохимические. Основная цель исследования заключалась в разработке методов прогнозирования механизмов раннего развития ОА.

Датасет состоит из широкого набора клинических данных и данных рентгенографических исследований, включая рентгенограммы фронтальной и боковой проекций коленей, фронтальной и боковой проекций бедер, а также (в ограниченном количестве) рентгенограммы кистей рук и (рис. 5). Доступны как данные первичного обследования, так и данные за второй, пятый, восьмой и десятый годы с момента первичного осмотра. Для каждого обследования, помимо набора рентгенограмм различных частей тела, присутствуют описания экспертов, включая степень ОА по шкале Келлгрена-Лоуренса, оценку сужения межсуставной щели, наличия остеофитов и т.п.



Рисунок 5 — Примеры данных из датасета CHECK, слева направо: передне-задняя проекция таза, боковая, фронтальная и аксиальная проекции коленного сустава, прямая проекция кистей

Для оценки особенностей ОА, при составлении датасета, эксперты опирались на атлас OARSI [4], при этом использовалась сокращенная шкала Келлгрена-Лоуренса (0–3 вместо 0–4). Каждая из рентгенограмм была оценена пятью независимыми экспертами. После исключения из датасета снимков с заведомо плохим качеством или снимков пациентов, у которых ранее был установлен искусственный сустав (см. рис. 6), распределение стадий ОА по шкале Келлгрена-Лоуренса (0–3) в датасете следующее — 1413 изображений для стадии 0, 2560 изображений стадии 1, 2173 для стадии 2, 167 для стадии 3.

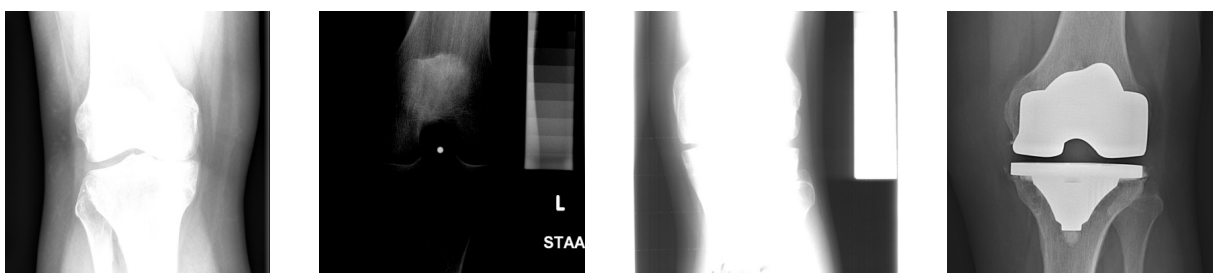


Рисунок 6 — Пример данных, исключенных из датасета CHECK

В ходе экспериментов было обнаружено, что имеющаяся разметка датасета имеет ошибки, которые сильно влияют на обучение автоматических классификаторов и не позволяют использовать ее непосредственно для разработки методов автоматической классификации ОА по шкале Келлгрена-Лоуренса. Поэтому в работе данный датасет также не используется.

1.3 Обзор современных подходов и идей в проектировании сверточных нейронных сетей

Первые работы по исследованию нейронных сетей появились еще в 80–90-х годах XX века, тогда же были предложены многие концепции и идеи. Было показано, что нейронные сети способны приближать практически любые функции, т.е. решать практически любые задачи. Тем не менее, недостаточные вычислительные мощности оборудования, а также отсутствие достаточного количества данных для обучения сетей не позволяли использовать нейронные сети для решения реальных задач. С развитием вычислительной техники и активным распространением интернета в начале 2000-х годов данные проблемы стало возможным постепенно решить, однако нейросети все равно оставались на заднем плане исследований.

Широко известными (и активно цитируемыми) нейронные сети стали в 2012 году, после победы в конкурсе по классификации изображений ImageNet, в котором различные методы соревновались собой в решении задачи классификации изображений на 1000 различных классов. Победителем стала сверточная нейронная сеть с архитектурой AlexNet, которая с большим отрывом победила традиционные методы вроде метода опорных векторов и т.п. С тех пор нейросети стали основной повесткой, и в литературе стало появляться все больше и больше материалов, посвященных решению разного рода задач при помощи них.

С момента первого успешного использования сверточной нейросети в конкурсе ImageNet до текущего момента точность классификации на ImageNet по метрике топ-5 ошибка классификации снизилась с 17% у AlexNet до 2,25% у SENet, что уже превышает точность классификации человеком, которая составляет примерно 5% [14]. При этом, точность работы для ансамблей сверточных

сетей (т.е. агрегации предсказаний нескольких классификаторов для получения финального ответа) еще выше.

1.3.1 AlexNet

Одной из первых сверточных нейронных сетей считают LeNet-5, предложенную в 1998 году в работе [15] и предназначенную для распознавания рукописных цифр на изображениях размера 30×30 . В этой архитектуре были заложены основы для последующих исследований в данном направлении и предложены схемы компоновки разного рода слоев, используемые и в современных сверточных сетях: чередование сверточных слоев со слоями пуллинга и использование нескольких полносвязных слоев на выходе сети.

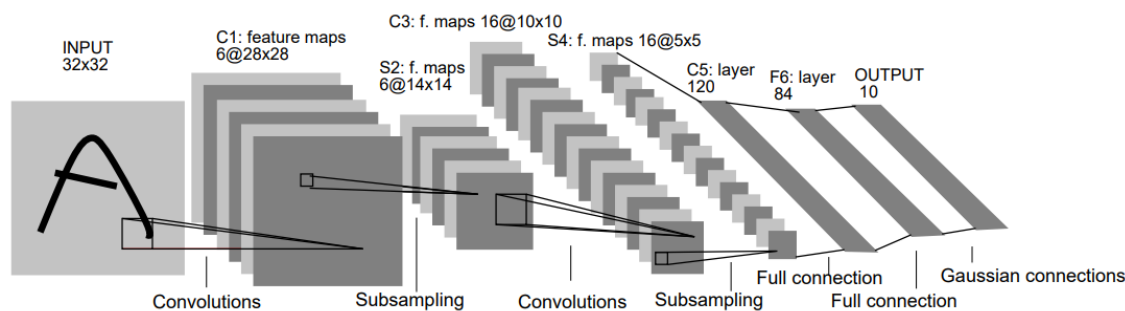


Рисунок 7 — LeNet-5

Ввиду имеющихся ограниченных вычислительных ресурсов в 1998 году, в качестве связи между сверточными слоями использовались не все возможные соединения нейронов, а специальная матрица разреженности, которая позволила существенно уменьшить количество вычислений. В качестве функции активации использовались два варианта: сигмоида, применяемая после последовательной пары сверточного слоя и слоя понижения размерности с ядром 2×2 , и масштабированный гиперболический тангенс для полносвязных слоев. Слой понижения размерности также имел обучаемые параметры, в отличие от более поздних слоев пуллинга. Переход от сверточных слоев к полносвязным организован через финальный сверточный слой с размером карты признаков $1 \times 1 \times 120$.

После исследований над LeNet-5 сверточные сети на долгое время выпали из поля зрения общественности, и вернулись к нему лишь в 2012 году, когда сверточная нейронная сеть AlexNet [16] сильно опередила традиционные методы

обработки изображений на задаче классификации в конкурсе ImageNet с ошибкой в топ-1 и топ-5, равной 37,5% и 17,0% против соответствующей ошибки в 45,7% и 25,7% у ближайшего по точности метода, основанного на использовании алгоритма SIFT [17].

AlexNet состоит из 8 слоев, 5 из которых сверточные, а 3 — полносвязные, и в сумме имеет 60 миллионов обучаемых параметров. В качестве функции активации — для ускорения процесса обучения — в данной архитектуре используется функция Rectified Linear Unit (ReLU [18]). В тестовом примере по обучению модели на датасете CIFAR-10 до ошибки классификации в 25% использование ReLU вместо тангенса позволило более чем в 6 раз ускорить процесс обучения [16].

В качестве слоя понижения размерности в данной архитектуре впервые был использован *слой перекрывающей субдискретизации* (pooling layer) по максимальному значению — т.е. слой субдискретизации, соседние фильтры которого частично обрабатывают общие группы пикселей предыдущего слоя из-за того, что шаг сдвига ядра меньше размера самого ядра, и для каждого ядра выбирают максимальное значение. Его использование позволило уменьшить ошибку топ-1 на 0,3%, а ошибку топ-5 — на 0,4%. Кроме того, авторы отмечают, что в случае использования слоя перекрывающей субдискретизации модель была менее подвержена переобучению [16]. Далее в работе, при упоминании слоя субдискретизации, будет подразумеваться субдискретизация по максимальному значению, если не сказано иное.

Также для уменьшения переобучения в работе использовалась аугментация данных в виде выбора случайной области 224×224 из исходного изображения размера 256×256 и, опционально, отражения по горизонтали, что позволило увеличить размер обучающей выборки в 2048 раз [16]. Еще одной формой аугментации было использование трансформации интенсивности RGB-каналов изображения и техника dropout [19], заключающаяся в обнулении нейронов в первых двух полносвязных слоях с вероятностью 0,5 (или 50%) на каждом проходе. После обнуления такие нейроны не вносят никакого вклада в прямой проход данных по сети, и не влияют на обратный проход (подсчет градиента).

К улучшениям AlexNet можно отнести использование ReLU в качестве функции активации, использование техники дропаута, перекрывающего пуллинга, а также более глубокая и широкая структура сети по сравнению с LeNet [20].

Это все стало возможным благодаря развитию вычислительной техники и увеличению ее мощности.

1.3.2 ZFNet, VGG

После успеха сети AlexNet в конкурсе ImageNet в 2012 году были приняты различные попытки улучшить оригинальную архитектуру.

Так, в архитектуре ZFNet [21] было предложено уменьшить размер фильтра первого сверточного слоя с 11×11 до 7×7 , а также изменить шаг (stride) для этого фильтра (с 4 до 2), что позволило уменьшить ошибку *top-5 error* с 15,3% до 11,4%. Top-5 error — это метрика, при которой алгоритму разрешается сделать 5 предположений о принадлежности объекта правильному классу, и результат классификации считается ошибочным, если верный ответ среди пяти предположений отсутствует.

Одна из известнейших моделей сверточных сетей — модель VGG-16 [22]. Она была представлена в 2014 году и является улучшенной версией AlexNet, в которой большие фильтры (размера 11 и 5 в первом и втором сверточных слоях) заменены на несколько фильтров размера 3×3 , следующих один за другим. Последовательность небольших фильтров вместо одного фильтра большого размера — характерная особенность этой архитектуры. Это позволяет сохранить рецептивное поле одного нейрона, уменьшив при этом количество параметров для обучения. Во всех последующих архитектурах данное архитектурное решение используется повсеместно.

В некоторых конфигурациях также используются фильтры размера 1×1 — это своего рода линейная трансформация входных значений слоя. Для уменьшения разрешения здесь используются отдельные слои субдискретизации с размером окна 2×2 .

Авторами было показано, что увеличение глубины нейросети значительно повышает точность ее работы для самых разных задач, включая классификацию и локализацию. Так, при тестировании на ImageNet в задаче распознавания объектов на изображении модели VGG-16 удалось достичь точность 92,7% для топ-5 категорий — т.е. ошибка практически в два раза меньше таковой для сети AlexNet. Однако, глубокие сети требуют больших вычислительных ресурсов

для обучения — к примеру, VGG-16 обучалась на протяжении нескольких недель при использовании видеокарт NVIDIA TITAN BLACK, одной из самых производительных на тот момент, а при недостаточном количестве обучающих данных, или из-за каких-либо других особенностей, процесс обучения вообще может не дать результатов.

1.3.3 ResNet

При построении глубоких сетей исследователи сталкиваются со многими проблемами. Помимо проблемы переобучения (особенно при недостаточном количестве тренировочных данных), которую с переменным успехом решают путем использования техники dropout, при увеличении глубины возникает проблема затухающих градиентов [23], т.е. глубокие сети сложнее обучать. Также в некоторых работах [24; 25] было замечено, что на определенных наборах данных, при добавлении к сети дополнительных слоев точность сначала увеличивается, а затем быстро ухудшается. Снижение точности обучения показывает, что не все сети легко оптимизировать. Т.е. вопреки ожиданиям, которые были основаны на предыдущих наблюдениях, добавление новых слоев в некоторых случаях ухудшает точность работы сети (пример на рис. 8).

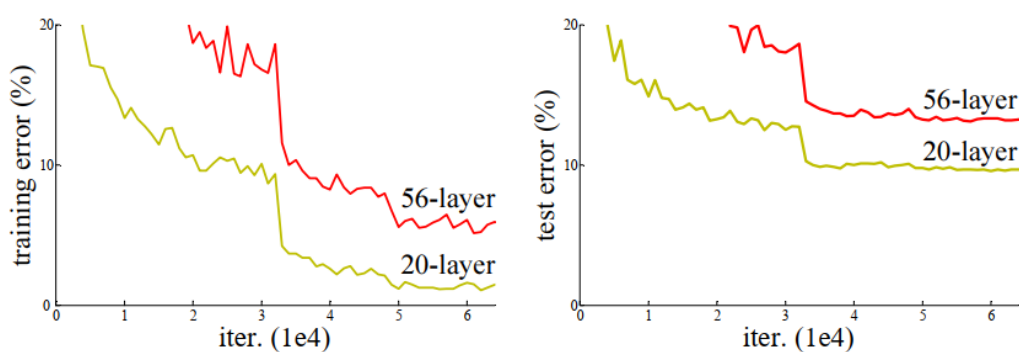


Рисунок 8 — Ошибка обучения (слева) и ошибка теста (справа) на CIFAR-10 с 20-ти и 56-ти слойными «простыми» сетями; более глубокая сеть имеет большую ошибку обучения и, следовательно, ошибку тестирования

Существуют различные подходы к решению проблемы затухающих градиентов — использование в качестве функции активации ReLU (как менее

страдающей от исчезающих градиентов, чем остальные функции), использование нормализации батчей [24] и др.

В работе [24] был предложен новый подход, который позволил тренировать очень глубокие сети — если в 2014 году одним из лидеров в решении задач классификации значилась сеть VGG-19, которая имела 19 слоев, то в 2015 году в конкурсе ImageNet победила сеть ResNet, основанная на новых идеях, которая имела 152 слоя. При этом авторы вдохновлялись философией архитектуры VGG, т.е. их сеть состоит из простых фильтров размера 3×3 , а при уменьшении размера слоя (обычно в два раза, после пулинга) количество фильтров на этом слое также увеличивается в два раза, чтобы сохранить обобщающие способности сети.

Основной идеей являлось использование для построения сети блоков, состоящих из двух слоев, и предсказывать этими двумя слоями не целевую функцию $H(x)$ для данной пары слоев (рис. 9, слева), а разницу между целевой функцией и входным вектором значений $F(x) = H(x) - x$, где x — входной вектор первого слоя блока (рис. 9, справа). Предполагается, что оптимизировать эту функцию гораздо легче, чем оригинальную $H(x)$. Более того — если в какой-то момент отображение будет оптимальным, то данный блок просто пропускает через себя входной вектор без изменений.

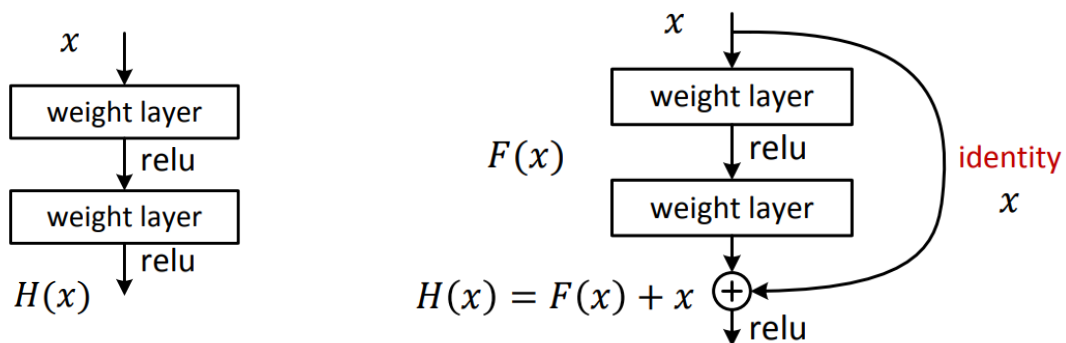


Рисунок 9 — Примеры пары слоев просто нейронной сети (слева) и блока в сети архитектуры ResNet (справа)

Основной блок новой архитектуры (рис. 9, справа) в базовом варианте состоит из двух слоев (либо сверточных, либо нет) и соединения быстрого доступа (shortcut connection), которое по сути является тождественным отображением — т.е. пропускает эти два слоя и «проталкивает» к выходу второго слоя блока вектор входных значений, в результате чего этот вектор просто добавляется к выходному значению всего блока.

В работе [24] авторы рассматривают несколько вариантов сетей с архитектурой ResNet, которые различаются структурой базовых блоков, из которых сеть и состоит, а следовательно — и глубиной. Так, для сети с глубиной 34 слоя используются базовые блоки, о которых было упомянуто выше — блоки, состоящие из двух слоев с неизменным количеством фильтров размером 3×3 (рис. 10, слева).

Для более глубоких модификаций подобного блока оказывается недостаточно, и вместо блока, состоящего из двух слоев, используются блоки, состоящие из трех слоев (т.н. «bottleneck block», рис. 10, справа). При этом, для упрощения вычислений, размерность блока сперва понижается при помощи фильтров размером 1×1 , а затем, после применения фильтров размером 3×3 меньшей размерности, чем на входе в блок, снова восстанавливается до исходной глубины. Это позволяет строить и успешно обучать сверхглубокие сети (глубиной 50, 101 и даже 152 слоя), которые показывают лучшие результаты, чем сети архитектуры VGG, при этом являясь вычислительно менее сложными из-за меньшего количества обучаемых параметров.

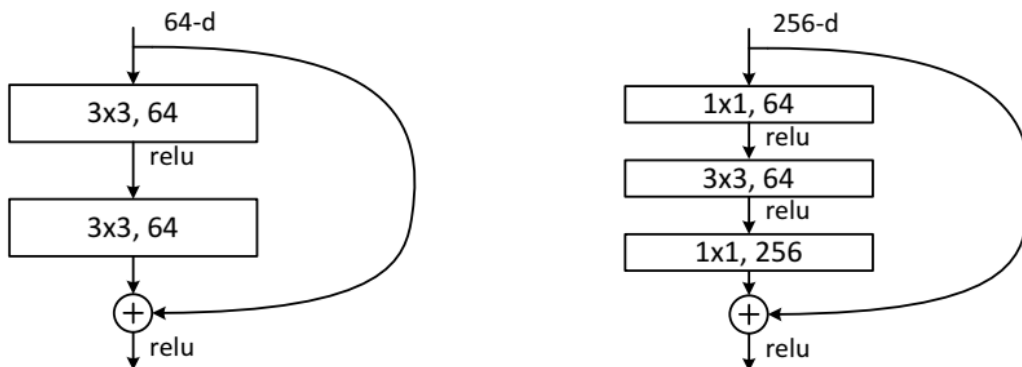


Рисунок 10 — Пример остаточной функции F из работы [24]: базовый блок для слоя размером 56×56 для ResNet-34 (слева) и расширенный, или bottleneck, блок для ResNet-50 и выше

Основные нововведения архитектуры ResNet — использование остаточного обучения и соединений быстрого доступа для увеличения глубины сети. Подробная структура моделей ResNet-18, ResNet-34 и ResNet-50 приведена в приложении А, в таблице 26.

1.3.4 Inception

Во многих работах было показано, что увеличение количества слоев сети позволяет повысить точность ее работы. Однако, при увеличении глубины растет количество обучаемых параметров сети, а следовательно — и требуемые для работы вычислительные ресурсы. При этом, если значительная часть параметров сети близка к нулю, то вычислительные ресурсы используются неэффективно. Поэтому задача увеличения эффективности работы нейронной сети не менее актуальна, чем увеличение точности ее работы.

Одним из основных путей решения описанных проблем является переход от плотно связанных слоев (когда все нейроны соединены друг с другом) к разреженным архитектурам, когда в сети присутствуют лишь значимые связи между нейронами (иначе говоря — к использованию разреженных матриц вместо простых матриц, представляющих собой веса слоев). К сожалению, на данном этапе развития техники, эффективная реализация подобных структур (умножение разреженных матриц в больших масштабах) невозможна, и даже эффект от стократного уменьшения количества параметров в случае разреженных связей будет нивелирован особенностями вычислительных комплексов и особенностями работы с памятью.

В работе [26] была предложена архитектура Inception, основанная на эффективном использовании имеющихся ресурсов и построении эффективного графа вычислений. Для этого предлагается строить сеть из специальных блоков (*Inception modules*), каждый из которых «сворачивает» входные данные свертками с ядрами различных размеров (1×1 , 3×3 и 5×5) на одном уровне, затем объединяет полученные результаты и подает их на вход следующему блоку. Свертки не меняют пространственную размерность данных (ширину и высоту слоя), поэтому такое объединение (простая конкатенация) возможно. Помимо свертки, в одном блоке также используется субдискретизация по максимуму, которая тоже не меняет пространственную размерность входных данных (рис. 11, слева). Последовательная комбинация таких блоков, перемежаемая слоями понижения пространственной размерности, и дает новую архитектуру сверточной нейронной сети.

Проблемой таких блоков является то, что для свертки с фильтром 5×5 , для входных данных большой глубины, вычисления могут быть слишком ре-

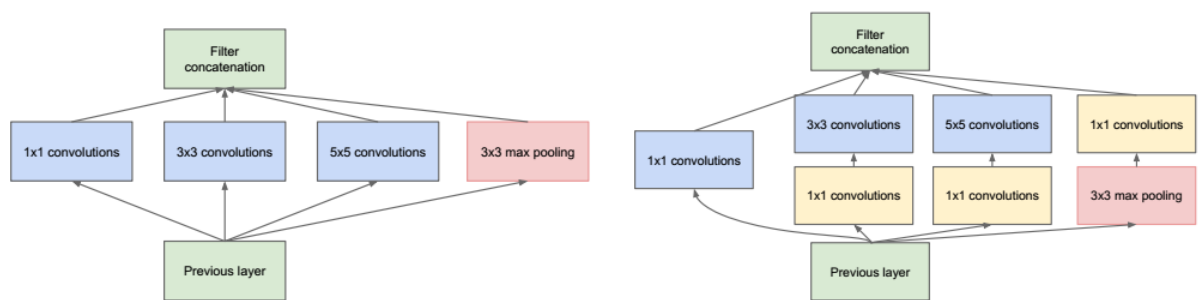


Рисунок 11 — «Наивная» версия блока Inception (слева) и блок с понижением размерности (справа) из одноименной архитектуры. Понижение размерности производится свертками 1×1

сурсоемкими. Для того, чтобы исправить подобную ситуацию, предлагается использовать блок Inception с понижением размерности (рис. 11, справа). Для этого входные данные сперва пропускаются через свертки размером 1×1 , которые понижают размерность входных данных (а именно глубину), а затем уже эти данные подаются на вход основным сверткам и слою субдискретизации. Одним из основных преимуществ подобной архитектуры (при использовании блоков, понижающих размерность) является то, что такой подход позволяет увеличивать общую глубину всей сети, не приводя при этом к существенным вычислительным затратам.

Первая сеть, построенная на описанных выше принципах, получила название GoogLeNet. В 2014 она заняла первое место в конкурсе ImageNet с результатом **6,67%** для метрики *top-5 error* при решении задачи классификации на 1000 классов. По сравнению с 2013 годом, точность классификации была практически удвоена. После конкурса, в работе [27], концепция Inception-блоков была расширена, и были предложены дополнительные способы уменьшения вычислительной сложности блоков.

Как было замечено выше, стандартные блоки Inception с фильтрами 5×5 , с ростом глубины входного тензора, становились все более вычислительно емкими. Для того, чтобы избежать этой проблемы, были предложены свертки 1×1 , понижающие размерность тензора по глубине. Однако, уменьшить вычислительную сложность для фильтра 5×5 можно и другим способом. Для этого одна свертка 5×5 заменяется на две последовательные свертки фильтрами 3×3 (рис. 12). Такая замена позволяет сохранить рецептивное поле и репрезентативную способность слоя, уменьшая при этом количество параметров свертки более чем в 2,5 раза.

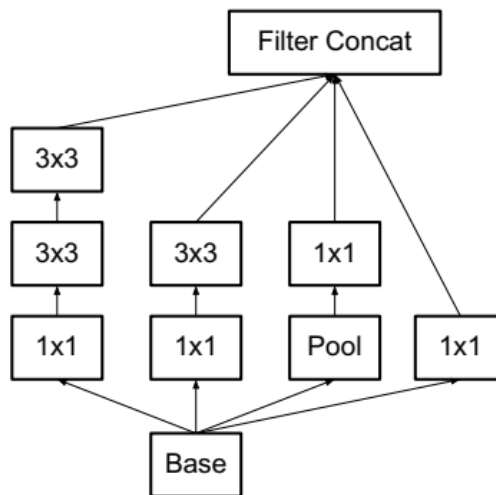


Рисунок 12 — Блок Inception, в котором свертка 5×5 заменена на две последовательные свертки 3×3

Далее авторы продолжают исследования по уменьшению пространственной агрегации путем замены фильтров $N \times N$ на последовательные фильтры $1 \times N$ и $N \times 1$ — т.е. пытаются использовать сепарабельные ядра фильтров. На практике было получено, что данная факторизация приводит к существенным улучшениям лишь на определенных слоях с фильтрами определенных размеров. Было показано, что хорошие результаты получаются при использовании последовательных сверток 1×7 и 7×1 (рис. 13).

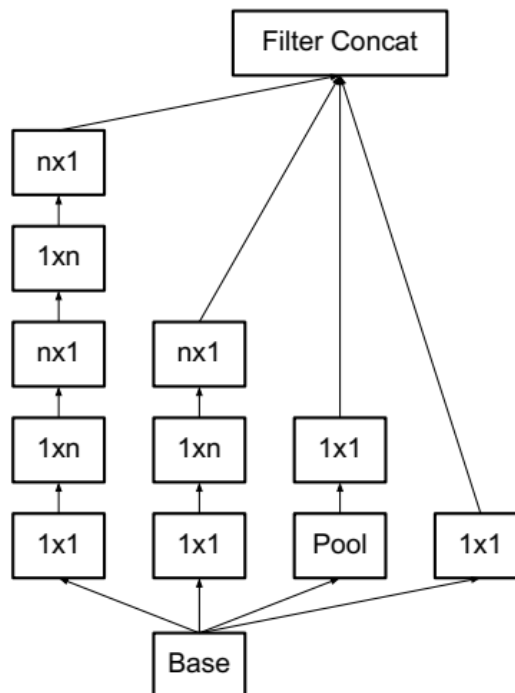


Рисунок 13 — Inception-блок после факторизации свертки $N \times N$

Чтобы избежать потери информации при уменьшении размерности в глубину, предлагается увеличивать вычислительные блоки «в ширину» — т.е. увеличивать количество сверток на одном уровне, а затем их конкатенировать. Пример такого вычислительного блока представлен на рис. 14.

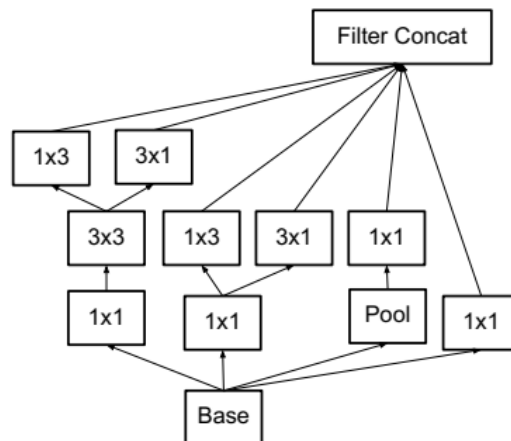


Рисунок 14 — Inception-блок с повышенным количеством сверток на одном уровне вычислений

Нейронная сеть, составленная из комбинации трех описанных выше блоков, была названа *Inception V2*. На данных конкурса ImageNet данной сети удалось достичь точности **5,6%** для top-5 error. Модификация данной сети, названная *Inception V3*, на том же датасете получила точность **4,2%** для top-5 error. В качестве модификации, для всех слоев (не только сверточных, но и полносвязных) была добавлена нормализация батчей, в блоках Inception были задействованы свертки 7×7 , факторизуемые в последовательность сверток 3×3 , а также был изменен способ обучения сети.

Таким образом, к основным улучшениям в Inception по сравнению с предыдущими архитектурами (AlexNet, VGG и др.) можно отнести использование блочной структуры сети и техники разделения и слияния (понижение и восстановление размерности) данных. Кроме того, в следующих версиях Inception (Inception V3 и др.) использовалась идея замены больших фильтров серией маленьких внутри блоков.

1.3.5 Inception-ResNet

Архитектуры ResNet и Inception были представлены примерно в одно время, и обе достигали примерно одинаковой точности работы, опережая своих предшественников. При этом, каждая из архитектур привнесла новые идеи в теорию глубоких сверточных сетей. Так, предложенные в ResNet соединения быстрого доступа (shortcut connections) позволили обучать очень глубокие модели, а способ построения модели из отдельных блоков с понижением размерности из Inception позволил сетям достигать высокой точности, оставаясь при этом эффективными в вычислительном плане. В 2016 году была опубликована работа [28], в которой рассматривалось объединение принципов обеих архитектур для получения более глубоких и эффективных моделей. Новая архитектура получила название Inception-ResNet.

В базовом варианте за основу новой архитектуры была взята архитектура Inception V3. Напомним, что архитектура Inception состоит из последовательностей блоков Inception трех типов, названных Inception-A, Inception-B и Inception-C, где каждый из типов блоков используется на своем уровне сети — в начале, середине и конце. Для комбинации с ResNet были использованы упрощенные версии оригинальных блоков Inception V3, в которые были добавлены соединения быстрого доступа (рис. 15).

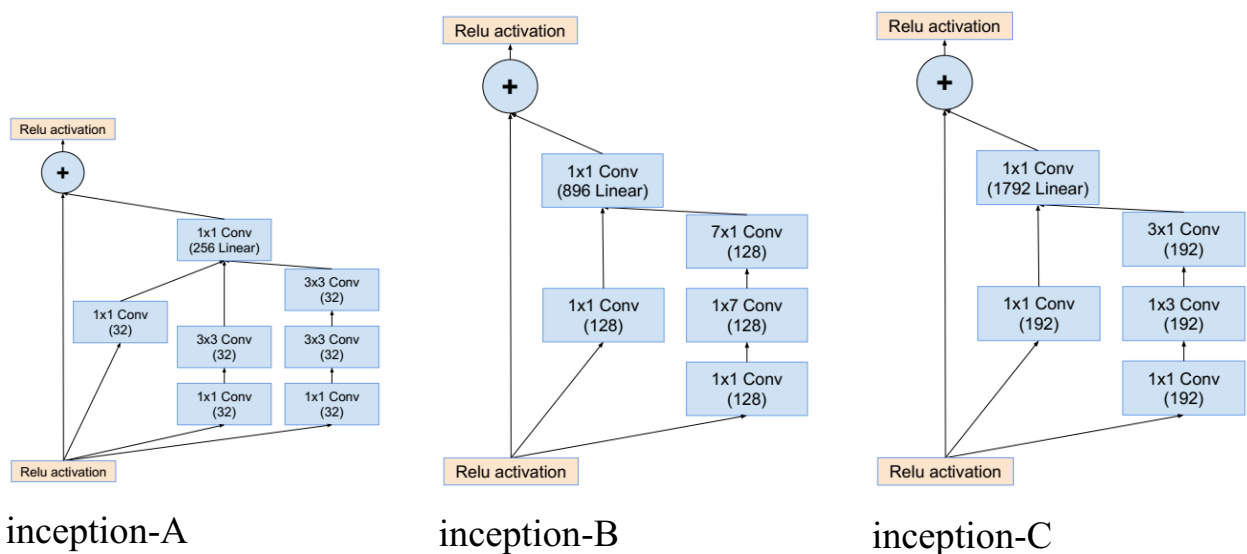


Рисунок 15 — Модули Inception-A, Inception-B и Inception-C для Inception-Resnet

Для этого требовалось, чтобы входные и выходные данные после сверток имели ту же самую размерность. Учитывая, что свертки в оригинальных блоках

Inception могут увеличивать количество каналов данных на выходе, в модифицированные блоки, после всех базовых сверток, были добавлены свертки 1×1 , которые позволили выравнивать глубину входа и выхода сети. При этом, слои субдискретизации из новых блоков Inception были удалены и их место заняли соединения быстрого доступа.

После добавления соединений быстрого доступа появилось ограничение на то, что вход и выход подобных блоков должны иметь одинаковую размерность. Но в таком случае необходим инструмент, позволяющий изменять размерность данных (как пространственную, так и в глубину). Для этих целей был введен специальный *редукционный блок* (reduction block), который на верхних своих слоях выполняет свертки с шагом, равным 2, что позволяет понижать вдвое пространственную размерность данных (рис. 16). Соединения быстрого доступа при этом не используются.

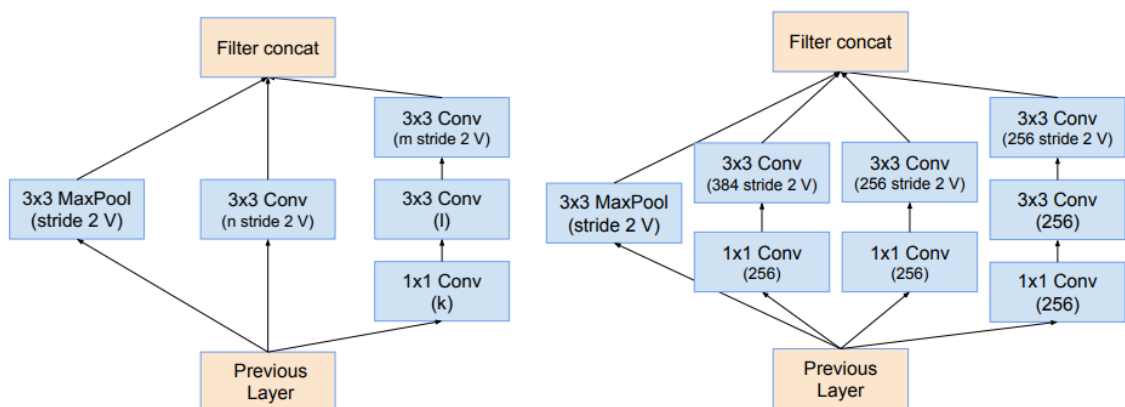


Рисунок 16 — Редукционные блоки для архитектуры Inception-ResNet, понижающие вдвое пространственную размерность данных. Разные блоки используются на разных уровнях сети

Как и в традиционной Inception-V3, в начале сети используются последовательность из нескольких обычных сверточных слоев в сочетании со слоями пуллинга, чтобы уменьшить начальную размерность данных перед их подачей в Inception-блоки (рис. 17). Для перехода от сверточных к полносвязным слоям используется глобальный усредняющий пулинг.

В работе отмечается, что использование остаточных связей позволяет существенно ускорить тренировку моделей архитектуры Inception. Кроме того, точность Inception-сетей, использующих остаточные связи, также превосходит точность сетей, эти связи не использующих.

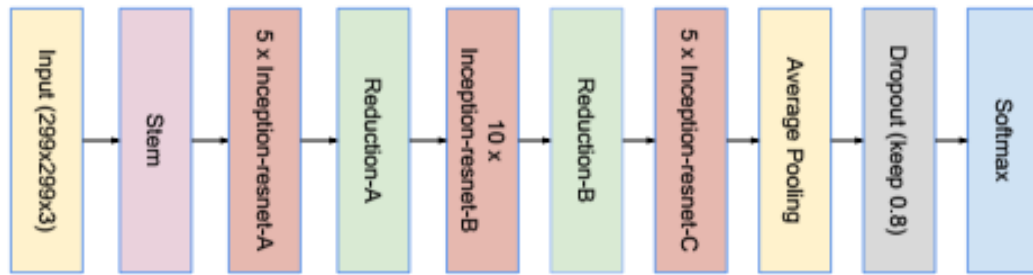


Рисунок 17 — Иллюстрация базовой версии архитектуры Inception-Resnet-v1 из работы [28]. Блок *Stem* обозначает последовательность из нескольких сверток с пуллингом, понижающих начальную размерность $299 \times 299 \times 3$ до $35 \times 35 \times 256$ на выходе из блока. Reduction-A, Reduction-B — различные вариации редукционных блоков

1.3.6 Xception

Стандартный сверточный слой пытается обучить фильтры для фильтрации входных данных в трехмерном пространстве с двумя пространственными размерностями и глубиной по количеству каналов, при этом обработка данных происходит одновременно по всем размерностям.

Основной идеей модуля Inception было упрощение (как в вычислительном плане) процесса свертки при помощи явного разделения пространственной и межканальной корреляции на серию независимых операций с последующим объединением результатов в выходной набор данных. Для этого сначала при помощи сверток размером 1×1 выполнялось разделение входных данных на несколько (3 или 4) пространств с меньшей размерностью по глубине, затем над каждым из полученных пространств выполнялась межпространственная корреляция (при помощи сверток 3×3 или 5×5), и в итоге полученные наборы данных, имеющие одинаковую пространственную размерность, объединялись при помощи конкатенации данных в пространстве каналов (рис. 11, справа).

В работе [29] авторы предлагают рассмотреть упрощенную версию модуля Inception (рис. 18, а), отбросив блок субдискретизации. В такой вариации блок может быть реорганизован в один большой слой свертки 1×1 , за которым следуют три свертки 3×3 , каждая из которых оперирует неперекрывающейся группой слоев свертки 1×1 (рис. 18, б). После данной переформулировки авторы задаются вопросом — какое именно количество сегментов 3×3 будет наиболее эффек-

тивным, и предлагают использовать *экстремальную* версию Inception-модуля, исходя из предположения, что пространственная и межканальная корреляция может обрабатываться полностью раздельно.

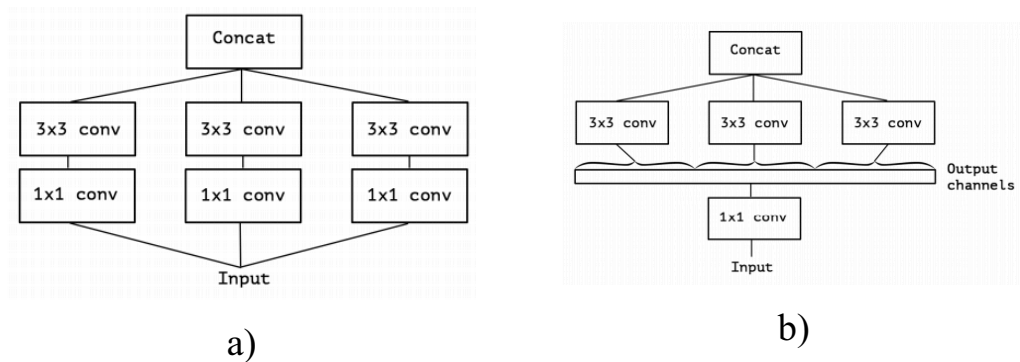


Рисунок 18 — Упрощенная версия Inception-модуля (a) и эквивалентная ей реорганизованная версия (b)

Экстремальный Inception-модуль состоит из одного слоя свертки с фильтром 1×1 , который призван обрабатывать межканальную корреляцию, и набора сверток 3×3 , каждая из которых обрабатывает свой канал, являющийся выходом первого сверточного слоя (рис. 19). По своей сути данная конструкция практически идентична разделимым по глубине сверткам из работы [30], за исключением двух моментов — порядка операций и использования нелинейности после каждой из сверток. Так, в разделимых по глубине свертках сначала, как правило, выполняется пространственная корреляция, а затем межканальная, а в Inception пространственная корреляция идет после межканальной. Более важным отличием является то, что в Inception каждая из операций свертки сопровождается нелинейностью в виде ReLU, в то время как в разделимых по глубине свертках нелинейность обычно отсутствует.

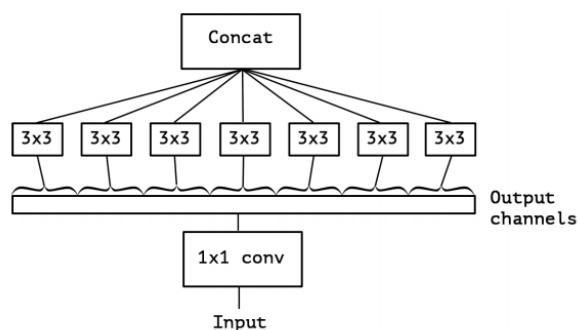


Рисунок 19 — Экстремальная версия Inception-модуля

На основе озвученных идей предлагается сеть **Xception** (Extreme Inception), которая по количеству обучаемых параметров схожа с сетью Inception

V3 и незначительно опережает ее в точности классификации на датасете ImageNet. Кроме того, помимо разделения сверток, в экстремальном Inception-модуле предлагается использовать соединения быстрого доступа из ResNet [24], что также позволило повысить точность работы.

Если сравнить Xception (22 910 480 параметров, точность на ImageNet 79%) с архитектурой InceptionResNet V2 [28] (55 873 736 параметров, точность 80,4%), которая довольно схожа с Xception после добавления к ней соединений быстрого доступа, то получится, что Xception имеет в два раза меньшее количество параметров (т.е. выигрывает по количеству требуемых ресурсов как для обучения, так и для использования), и при этом лишь незначительно уступает в точность Inception-ResNet.

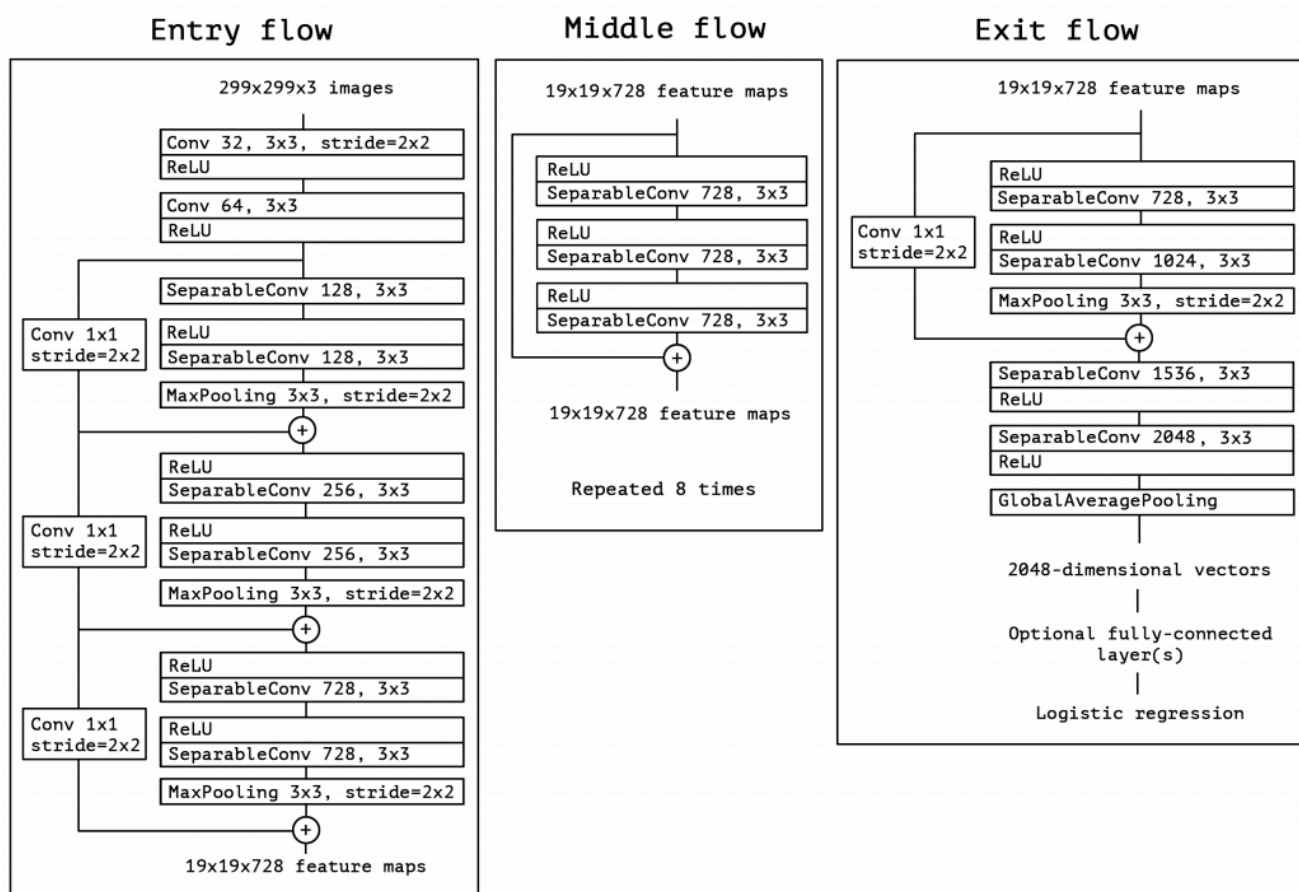


Рисунок 20 — Внешний вид базовой архитектуры Xception

Общая схема классической архитектуры Xception представлена на рис. 20. Сначала данные проходят через Entry flow, затем через Middle flow, набор слоев которого повторяется 8 раз, и затем через Exit flow, на выходе которого находится полносвязный слой с числом выходов, равным количеству классов датасета. Отметим, что для всех сверточных блоков (Convolution, SeparableConvolution) применяется нормализация батчей.

Преимуществом Xception по сравнению с предшественниками можно отнести меньшее количество параметров при сопоставимой точности работы за счет разделения стандартных сверток на пространственную и свертку по глубине.

1.3.7 ResNeXt

Дизайн архитектур нейросетей становится все более трудоемким с ростом количества гиперпараметров (ширины сети, размера фильтров и т.п.). В архитектуре VGG [22] был предложен простой способ построения глубоких архитектур нейронных сетей — использование серии сверточных слоев с одной и той же структурой, который затем был расширен в сетях архитектуры ResNet [24], где было предложено использовать не серии слоев, а серии похожих строительных блоков из нескольких сверточных слоев (с одинаковой топологией).

В свою очередь, архитектуры Inception для достижения лучшей точности больше опираются на дизайн нейронной сети, нежели на количество слоев, и в качестве основной идеи конструирования строительных блоков придерживаются стратегии *разделения-преобразования-агрегации* — т.е. в блоке сначала происходит понижение размерности входных данных фильтром 1×1 , затем преобразование набором фильтров 3×3 , 5×5 и т.п., и наконец происходит объединение результатов разных веток в один выходной вектор при помощи конкатенации. Эта стратегия позволяет повысить репрезентативную способность строительных блоков, уменьшая при этом их вычислительную сложность, однако имеет и свои нюансы — неочевидность выбора размера фильтров, структуры блоков и т.п.

В работе [31] предлагается архитектура, названная **ResNeXt**, которая стремится объединить плюсы архитектур ResNet (серии строительных блоков с соединениями быстрого доступа) и Inception (стратегию разделения-преобразования-агрегации), при этом устранив неочевидность выбора деталей построения базовых блоков сети.

Для этого предлагается использовать блоки с соединениями быстрого доступа, которые состоят из нескольких веток. Количество этих веток называется *кардинальностью* (cardinality) блока. Каждая из них сначала сильно понижает

размерность, затем выполняет преобразования над данными и восстанавливает размерность до исходной, а результаты перед применением соединений быстрого доступа агрегируются при помощи суммирования (это возможно, т.к. все ветки имеют одинаковую топологию). Существуют различные эквивалентные формулировки подобных блоков (их подробное описание можно найти в [31]), однако базовый вид блока приведен на рис. 21, справа.

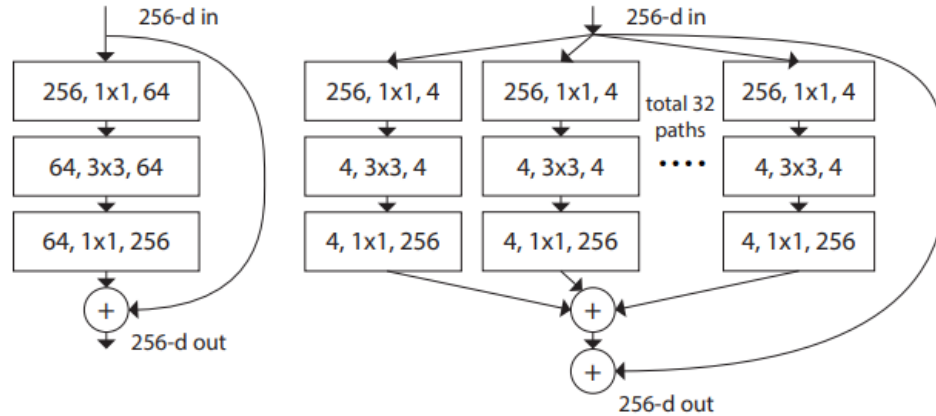


Рисунок 21 — Иллюстрация базового блока ResNet (слева) и базового блока ResNeXt с кардинальностью 32 (справа)

По утверждению авторов, увеличение кардинальности блока является более эффективным способом повышения точности работы нейронной сети, нежели увеличение ее глубины или ширины. В тестах на датасете ImageNet ResNeXt превосходит архитектуры ResNet-101/152 [32], ResNet-200 [33], Inception-v3 [27] и Inception-ResNet-v2 [28] — в частности, 101-слойная ResNeXt достигает лучшей точности, чем ResNet-200, обладая при этом в 2 раза меньшей вычислительной сложностью. При этом, ResNeXt имеет более простую архитектуру, чем все разновидности Inception (приложение А, табл. 27).

Основные идеи архитектуры ResNeXt — использование кардинальности вычислительного блока для управления его сложностью, однообразная топология и группировка сверток.

1.3.8 DenseNet

Рассмотрим изображение x_0 , которое подается на вход нейронной сети, состоящей из L слоев, каждый из которых выполняет некоторую нелинейную

трансформацию H_l , состоящую из комбинации сверток, слоев субдискретизации, нормализации батчей [24], ReLU и т.п. Если обозначить выход l -го слоя сети как x_l , то в традиционной сети переход от $(l - 1)$ -го слоя к l -му может быть обозначен как $x_l = H_l(x_{l-1})$.

Во многих исследованиях показано, что сверточные нейронные сети могут быть очень глубокими, более точными и эффективными в плане использования ресурсов, если они используют соединения быстрого доступа между слоями [24; 28]. Однако, сети из семейства ResNet ограничиваются соединениями быстрого доступа лишь между парой соседних сверточных слоев. Используя обозначения выше, соединения быстрого доступа могут быть обозначены следующим образом:

$$x_l = H_l(x_{l-1}) + x_{l-1}.$$

В [34] предлагается расширить использование подобных соединений и добавить еще больше дополнительных связей между слоями. В этой работе описывается новая архитектура DenseNet (Dense Convolutional Network), отличительной особенностью которой является то, что все слои в строительных блоках сети соединены со всеми последующими слоями блока:

$$x_l = H_l([x_0, x_1, \dots, x_{l-1}]),$$

где $[x_0, x_1, \dots, x_{l-1}]$ является объединенным вектором выхода слоев с номерами $0, 1, \dots, l - 1$. Если обозначить через L количество соединений в традиционной сверточной сети, где каждый слой связан только с соседним слоем, то в предложенной архитектуре слоев будет $\frac{L(L+1)}{2}$. В качестве функции H_l в DenseNet выступает комбинация из сверточного слоя с фильтром размера 3×3 , нормализация батчей и ReLU.

Таким образом, по аналогии с ResNet, которая состоит из блоков, каждый из которых, помимо различных преобразования между своими слоями, «проталкивает» вектор входных значений на выход блока и конкатенирует их с результатом внутриблоковых вычислений, DenseNet также состоит из набора блоков по несколько сверточных слоев каждый. Для того, чтобы передавать максимум информации, в то же время сохраняя линейную структуру сети, каждый слой блока имеет дополнительные связи со всеми предыдущими слоями (получает дополнительные входные данные от всех предыдущих слоев) и передает

свои собственные выходные значения всем последующим слоям (рис. 22). Такой блок называется *плотно-связанным* (*dense*) блоком. Важным отличием плотно-связанного блока от базового блока ResNet является то, что карты признаков не суммируются поэлементно, прежде чем передать их в следующий блок, а комбинируются посредством конкатенации.

Ввиду того, что внутри плотно связанного блока между слоями используется конкатенация карт признаков, изменение пространственной размерности данных представляется невозможным, поэтому блочная структура необходима, чтобы была возможность гибко менять размерность данных при помощи специальных переходных слоев (*transition layer*). Переходный слой состоит из свертки 1×1 , которая используется для изменения количества каналов в данных, и слоя субдискретизации с усреднением (*average pooling*) 2×2 , который позволяет регулировать пространственную размерность данных. Кроме того, каждый переходный слой использует нормализацию батчей.



Рисунок 22 — DenseNet с двумя плотно-связанными блоками. Слои между двумя смежными блоками называются переходными слоями и меняют размеры карты признаков посредством свертки и пуллинга. В качестве перехода от сверточных слоев к полносвязному используется GAP

По заявлениям авторов, из-за особой структуры связей между слоями, сети DenseNet позволяют достигать точности, сравнимой с лидерами — а на некоторых задачах даже их превосходит. Эффективное использование информации позволяет DenseNet получать те же репрезентативные способности, что и у более глубоких сетей, обладая при этом меньшим количеством параметров. Это происходит из-за того что, в отличие от простых сверточных сетей, здесь не приходится обучать избыточные карты признаков. Традиционные архитектуры рассматривают сеть как алгоритм с состоянием, которое передается от одного слоя к другому, т.е. каждый слой знает лишь предыдущее состояние системы. Однако, изменение состояния в предыдущих слоях также несет определенную информацию, которая в случае традиционных архитектур теряется.

Еще одним преимуществом DenseNet является простота обучения, т.к. каждый слой, из-за особенностей построения связей, имеет прямой доступ к градиентам из функции потерь. Это позволяет эффективнее решать проблему затухающих градиентов. Также в работе отмечается, что плотные соединения обладают обобщающим эффектом, который уменьшает переобучение на задачах с небольшим количеством тренировочных данных. Это означает, что DenseNet хорошо работает на маленьких датасетах. Учитывая, что в нашем случае датасет ОАИ имеет относительно небольшой размер, этот факт позволяет предположить, что удастся получить большую точность классификации ОА, нежели предыдущими методами [141].

В целом, отличием DenseNet от других архитектур является передача информации между блоками. Подробная структура сети DenseNet-121, а также комментарии об особенностях их реализации программными средствами, приведены в таблице 28 в приложении А.

1.3.9 Squeeze-and-Excitation Networks

Традиционным центральным блоком сверточной нейронной сети является оператор свертки, который позволяет формировать информативные карты признаков, учитывающие как пространственную, так и межканальную информацию. В работе [35] делается акцент на отношениях между разными каналами одного слоя. Авторы предлагают новый элемент архитектуры, называемый блоком сжатия и возбуждения (Squeeze-and-Excitation block, SE), который позволяет усилить обобщающую способность нейронной сети или ее части путем явного моделирования взаимозависимости между различными слоями сверточного блока. Предложенный механизм позволяет выполнять рекалибровку выходных данных сверточного блока (иначе говоря — ставить в соответствие каждому слою его вес), обучаясь использовать глобальную информацию для акцентирования внимания сети на информативных признаках, в то же время подавляя признаки неинформативные. Структура блока сжатия и возбуждения представлена на рис. 23.

Сначала для карты признаков U выполняется операция сжатия F_{sq} , которая позволяет получить дескрипторы каждого канала путем агрегирования

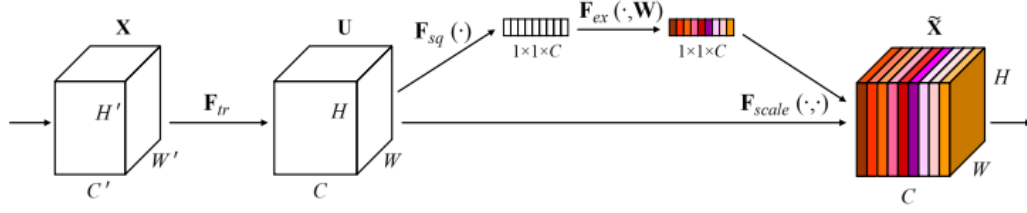


Рисунок 23 — Иллюстрация блока сжатия и возбуждения

содержимого каждого слоя в одно число при помощи операции глобального усредняющего пуллинга:

$$\mathbf{z}_c = \mathbf{F}_{sq}(u_c) = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W u_c(i, j),$$

где $\mathbf{z}_c$ — дескриптор канала c ($\mathbf{z} \in R^C$), u_c — содержимое слоя для канала c (изображение размера $H \times W$).

Для полученного таким образом вектора дескрипторов выполняется процедура адаптивной рекалибровки или *возбуждения* ($\mathbf{F}_{ex}$). Этот механизм принимает на вход вектор дескрипторов $\mathbf{z}_c$ и состоит из двух полносвязных слоев, первый из которых обладает нелинейностью в виде ReLU, а второй — нелинейностью в виде сигмоиды. При этом, первый полносвязный слой уменьшает размерность вектора на некоторый коэффициент r , а второй слой восстанавливает размерность до исходной — это позволяет гибко управлять сложностью и обобщающей способностью всего SE-блока. Если обозначить сигмоиду через σ , а нелинейность ReLU через δ , то операция возбуждения записывается следующим образом:

$$\mathbf{s} = \mathbf{F}_{ex}(\mathbf{z}, \mathbf{W}) = \sigma(\mathbf{W}_2 \delta(\mathbf{W}_1 \mathbf{z})),$$

где $\mathbf{W}_1 \in R^{\frac{C}{r} \times C}$, $\mathbf{W}_2 \in R^{C \times \frac{C}{r}}$. В качестве функции активации для второго полносвязного слоя блока используется сигмоида, как показавшая наилучшую эффективность для операции возбуждения в оригинальной работе [35]. Изменение размерностей на коэффициент r необходимо для того, чтобы обеспечить обучение блока нелинейным зависимостям между каналами, при этом r позволяет управлять количеством параметров и вычислительной сложностью SE-блока. В работе показано, что увеличение сложности блока не является прямо пропорциональным улучшению производительности блока, однако маленькие значения уменьшения сильно увеличивают количество обучаемых параметров сети. В

качестве базового значения коэффициента уменьшения авторами предложено использовать число 16, что является балансом между сложностью блока и его производительность, однако утверждается, что в некоторых ситуациях для разных уровней нейронной сети, вероятно, потребуется использовать различные значения коэффициента уменьшения r .

Финальный выходной набор данных блока $\tilde{\mathbf{X}}$ получается после выполнения операции рекалибровки карты признаков $\mathbf{U}$ с учетом вектора коэффициентов $\mathbf{s}$:

$$\tilde{\mathbf{x}}_c = \mathbf{F}_{scale}(u_c, \mathbf{s}_c) = u_c \mathbf{s}_c,$$

где при помощи $\mathbf{F}_{scale}(u_c, \mathbf{s}_c)$ обозначено поэлементное перемножение карты признаков $u_c \in R^{H \times W}$ и соответствующего ей скалярного коэффициента $\mathbf{s}_c$.

Базовое применение SE-блоков реализовано авторами в архитектуре SENet-154, которая основана на модификации ResNeXt-152 $64 \times 4d$. В качестве изменений можно назвать уменьшение каналов в сверточных слоях понижения межканальной размерности внутри базовых блоков вдвое, замену первого сверточного слоя с фильтрами 7×7 на три последовательных сверточных слоя с фильтрами 3×7 и добавление дропаута (с коэффициентом 0,2) перед финальным полносвязным слоем, отвечающим за классификацию. Из-за отсутствия технической возможности обучения сетей с очень глубокой архитектурой в рамках текущего исследования данная архитектура в работе не используется.

Однако, из-за своей простой структуры и легковесности, SE-блоки могут быть использованы для улучшения уже известных предобученных нейронных сетей различных архитектур, позволяя в большинстве случаев повысить итоговое качество их работы путем простого добавления соответствующих блоков в уже готовые архитектуры (в том числе и для случая переноса обучения), не приводя при этом к существенному повышению вычислительной сложности.

Ниже приведены архитектурные решения по интеграции блоков сжатия и возбуждения в архитектуры Inception, ResNet, ResNeXt, DenseNet и Xception, которые были апробированы на задаче классификации остеоартрита коленного сустава в рамках диссертации [142].

SE-InceptionV3

В случае с архитектурой Inception используется схема из оригинальной работы, которая заключается в размещении SE-блоков непосредственно после каждого Inception-блока (рис. 24).

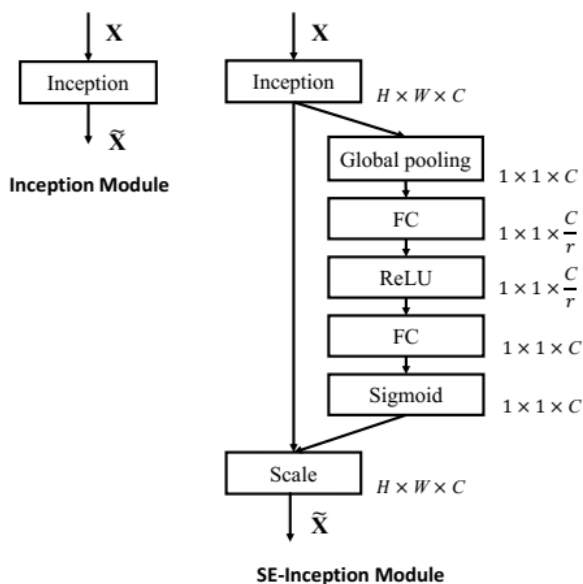


Рисунок 24 — Интеграция SE-блоков в архитектуру Inception: исходный модуль (слева) и модуль с добавленным блоком сжатия и возбуждения (справа); в качестве коэффициента редукции r использовалось значение 16

Подобная схема расположения позволяет использовать уже готовые, обученные модули Inception для применения на новой задаче — в данном случае, на задаче автоматической классификации остеоартрита. Принципиальной разницы в размещении SE-блоков в данном случае нет — при расположении данных блоков перед каждым Inception-блоком, а не после, ситуацию можно рассматривать как расположение этих блоков после предыдущего блока.

В [35] приводятся рассуждения о возможности более глубокой интеграции SE-блоков внутри готовых вычислительных модулей сверточных сетей, в том числе и Inception, путем размещения этих блоков непосредственно после сверточных слоев, однако подобное архитектурное решение мало того, что требует серьезной модификации структуры модели, но и за счет повышения количества SE-блоков приводит к неизбежному увеличению вычислительной сложности моделей, поэтому данное направление исследований в текущей работе не рассматривается.

SE-ResNet, SE-ResNeXt

Для удобства реализации, а также для полного соответствия базовой идее из работы [35] в архитектурах ResNet и ResNeXt блоки сжатия и возбуждения были интегрированы между финальным сверточным слоем модуля и соединением быстрого доступа, т.е. использовалась стратегия размещения SE-PRE из исходной работы. В этой схеме рекалибровка весов выполняется до операции суммирования между выходными и входными данными residual-блока. Данная схема в экспериментах показала наиболее высокие результаты производительности. Подробная структура новых строительных блоков для SE-ResNet и SE-ResNeXt представлена на рис. 25.

Эксперименты показывают, что наибольшее улучшение производительности моделей достигается использованием SE-блоков в каждом из вычислительных модулей этих моделей, не приводя при этом к существенному повышению вычислительной сложности и количества обучаемых параметров.

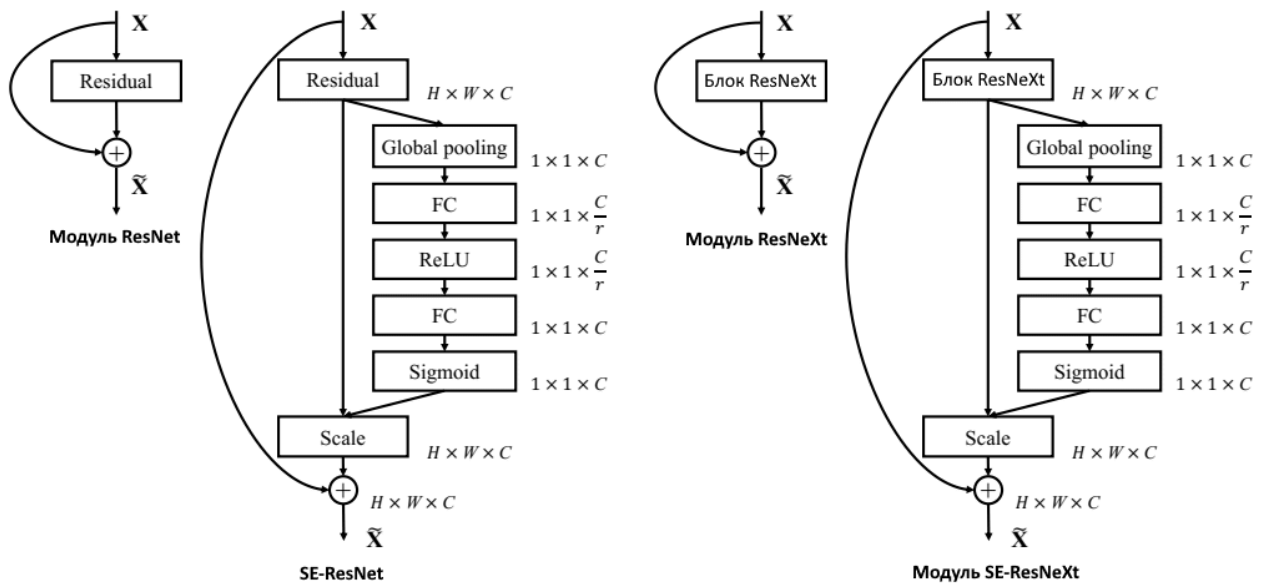


Рисунок 25 — Интеграция SE-блоков в архитектуры ResNet и ResNeXt: исходный модуль (слева) и модуль с блоком сжатия и возбуждения (справа)

Подробная структура моделей упомянутых архитектур ResNet и ResNeXt различной глубины с интегрированными блоками сжатия и возбуждения представлена в таблице 29 приложения А.

SE-Xception

Состоящая из большого количества архитектурных решений схема Xception открывает широкий простор для применения блоков сжатия и возбуждения. Помимо традиционных вариантов вроде расположения SE-блоков перед основными строительными блоками Xception (стратегия SE-PRE) или после них (стратегия SE-POST), возможны и другие варианты. В частности, в работе [36], посвященной автоматической классификации заболеваний растений по повреждениям листьев, предлагается вариант с расположением SE-блоков между слоями пространственной ($3 \times 3 \times 1$) и межканальной ($1 \times 1 \times N$) сверток, что по заявлениям авторов позволяет «усилить» разделимые по глубине сверточные блоки.

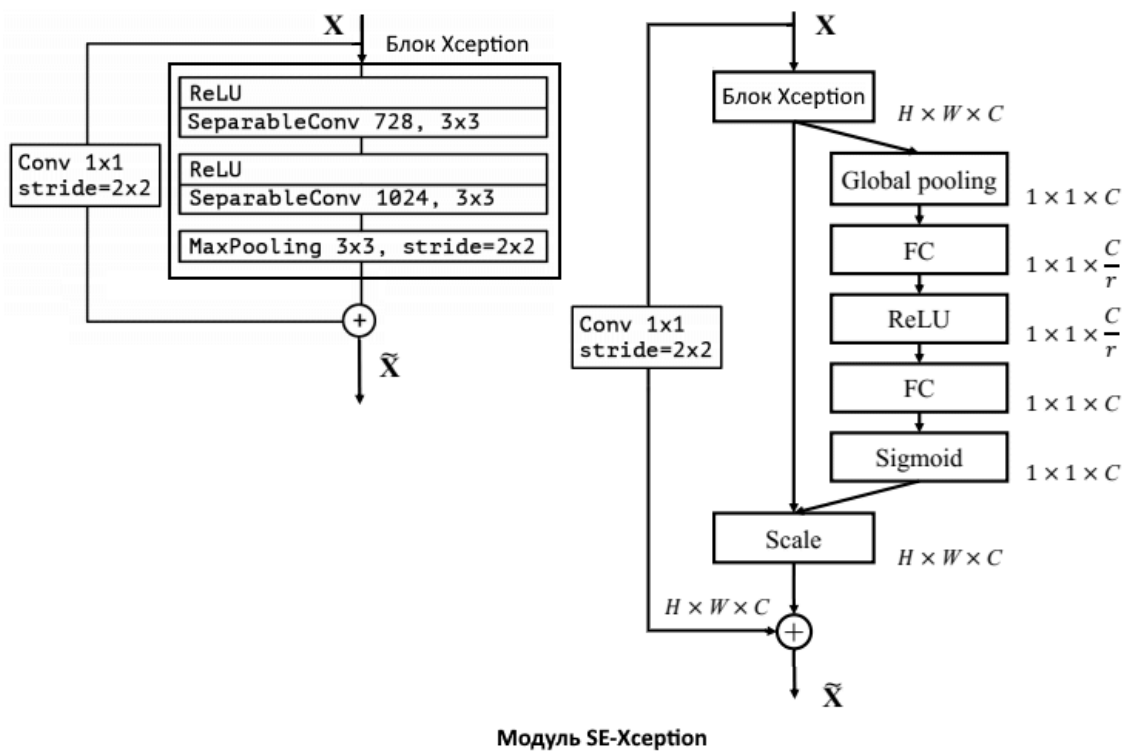


Рисунок 26 — Интеграция SE-блоков в архитектуру Xception: исходный модуль (слева) и модуль с добавленным блоком сжатия и возбуждения (справа)

В работе [35] показано, что в целом место расположения SE-блоков не принципиально, и их эффект устойчив к изменению положения внутри вычислительного блока нейронной сети (в качестве таких блоков выступают inception-блоки, отдельные модули ResNet и т.п.). В связи с этим, для модификации моделей Xception было принято решение реализовать стратегию SE-POST

размещения блоков сжатия и возбуждения из исходной работы. В данной стратегии блоки сжатия и возбуждения должны располагаться на выходе модуля, после срабатывания соединений быстрого доступа и т.п. Подобная схема незначительно теряет в производительности, однако более проста в реализации. Финальная схема полученного Xception-блока представлена на рисунке 26.

SE-DenseNet

В отличие от ResNet, где SE-блоки в стандартном варианте располагаются перед соединениями быстрого доступа, в архитектуре DenseNet с точки зрения реализации и гибкости SE-блок удобно размещать непосредственно перед основными сверточными слоями (или группами логически связанных слоев). Подробная схема отображена на рис. 27.

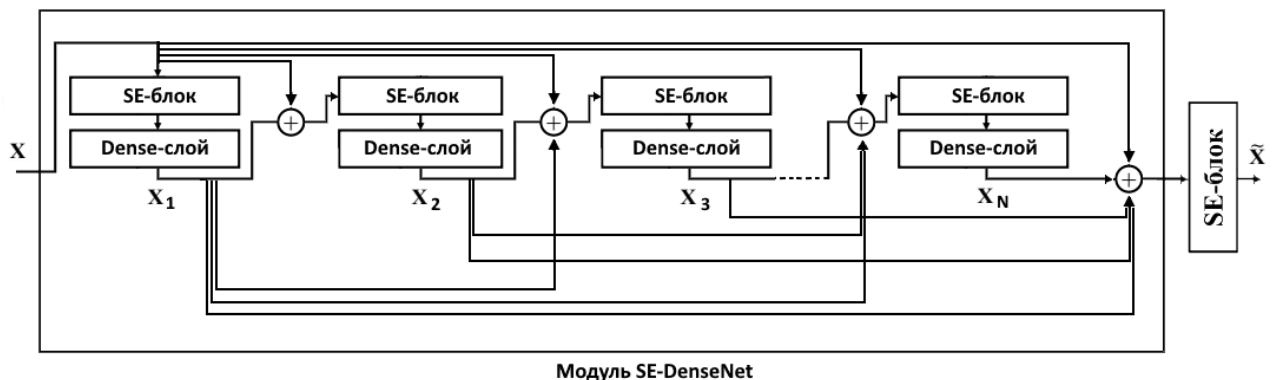


Рисунок 27 — Интеграция SE-блоков в архитектуру DenseNet на примере плотно-связанного блока

Учитывая, что SE-блоки в данной компоновке располагаются практически после каждой пары сверточных слоев в плотно-связанном блоке, подобный подход в большей степени соответствует идее глубокой интеграции SE-блоков в готовые архитектуры из [35]. Если в упомянутых ранее архитектурах это местами требовало серьезной переработки всей архитектуры, то в данном случае из-за особенностей построения плотно-связанных блоков, эта идея легко ложится на имеющуюся структуру без необходимости изменения всей структуры.

Подробная структура модели архитектуры DenseNet-121 с интегрированными блоками сжатия и возбуждения представлена в таблице 30 приложения А.

1.4 Обзор методов оценки точности локализации и классификации

1.4.1 Оценка качества локализации

Существуют различные метрики для оценки качества сегментации и локализации. К основным можно отнести метрику пиксельной точности (pixel accuracy), пересечение над объединением (intersection over the union, IoU), которое еще называют коэффициентом Жаккарда (Jaccard index) и мера Серенсена (еще известная в литературе как F-мера).

Пиксельная точность используется преимущественно в задачах семантической сегментации и означает процент пикселей, которые были классифицированы корректно. Однако эта метрика довольно грубая и плохо работает в случае несбалансированного размера областей различных классов (к примеру, при сегментации маленьких объектов на изображении).

Коэффициент Жаккарда является более универсальным и простым в вычислении критерием оценки. Применительно к задаче локализации он выражается следующим образом:

$$IoU = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|},$$

где в качестве области A выступает вручную аннотированный прямоугольник области интереса, в качестве области B — прямоугольная область, найденная исследуемым методом. Метрика принимает значения от 0 до 1, где 0 означает полное отсутствие пересечения предсказанной области с искомой, а 1 — полное совпадение этих областей, т.е. идеальный результат.

Мера Серенсена, она же F-мера — бинарная мера сходства, которая может быть определена как

$$F = \frac{2|A \cap B|}{|A| + |B|},$$

где $|A|, |B|$ — размер соответствующих множеств. Данная метрика коррелирует с коэффициентом Жаккарда и совпадает с ней в крайних положениях 0 и 1 при соответствующих условиях (полное отсутствие совпадения для 0 и полное совпадение для значения 1). Для пары этих метрик справедливо соотношение

$$F/2 \leq IoU \leq F.$$

Отличие этих двух метрик состоит в том, что мера IoU «наказывает» плохие результаты локализации при выводе численной оценки сильнее, чем мера Серенсена. В некоторых задачах данный факт может иметь большое значение. Тем не менее, для удобства сравнения результатов с другими методами локализации области коленного сустава, в дальнейшем в рамках данной работы будет использована метрика IoU.

1.4.2 Оценка качества классификации

Матрица ошибок и связанные с ней метрики

Существуют различные способы оценки производительности алгоритмов машинного обучения и производных от них классификаторов. Основная масса данных способов строится на использовании т.н. *матрицы неточностей* (confusion matrix), которая содержит в себе количество корректно и некорректно классифицированных примеров для каждого класса [37]. На рис. 28 приводится пример такой матрицы для случая бинарной классификации, где один класс трактуется как *позитивный* (P), а второй — как *негативный* (N).

	Positive(P)	Negative(N)
True(T)	True Positive (TP)	False Positive (FP)
False(F)	False Negative (FN)	True Negative (TN)
	P=TP+FN	N=FP+TN

Рисунок 28 — Иллюстрация матрицы неточностей для бинарной классификации

Зеленая диагональ матрицы содержит количество корректно предсказанных образцов для обоих классов, остальные значения — количество ошибочных

предсказаний. Так, если позитивный класс был верно предсказан позитивным, он обозначается True Positive (TP); если он был ошибочно распознан как негативный — то обозначается как False Negative (FN), или ошибкой второго типа [38]. Если образец негативного класса был корректно распознан негативным, то он обозначается True Negative (TN), а в случае ошибки, когда он был распознан как позитивный — False Positive (FP), или ошибка первого типа. Упомянутые значения матрицы ошибок используются для вычисления довольно большого количества метрик оценки классификатора.

Accuracy. Наиболее распространенным эмпирическим методом оценки точности классификации является процент корректно классифицированных примеров — *правильность* (или accuracy):

$$accuracy = \frac{TP + TN}{FP + FN + TP + TN}.$$

Данная мера не акцентируется на каком-то конкретном классе и оценивает ситуацию в целом, без детального анализа. В случаях, когда датасет не сбалансирован (т.е. число примеров в каждом классе значительно отличается), данная оценка может вводить в заблуждение.

К примеру, пусть нам необходимо выполнить бинарную классификацию на датасете, в котором один класс представлен 90 образцами, а второй содержит всего 10 примеров. Алгоритм может получить точность 90% просто если будет предсказывать все подаваемые ему на вход данные как принадлежащие классу 1, что, очевидно, некорректно. Это происходит из-за того, что мы не учитываем распределение тренировочных примеров каждого из классов. Кроме того, данный показатель не дает информации о том, где именно ошибается классификатор.

Последняя причина может быть очень важна в медицинских исследованиях. К примеру, здесь необходимо быть точно уверенным, чтоб классификатор не выдает ложноотрицательные (false negative) результаты — т.е. в том, что случай болезни не будет классифицирован ложно при ее наличии, т.к. это может привести к серьезным последствиям. В то же время меньше проблем возникнет в случае, если здоровый пациент будет классифицирован как имеющий болезнь — т.н. ложноположительная (false positive) классификация. В данном случае необходимо будет просто провести дополнительную проверку, и эта ошибка не приведет к серьезным последствиям.

Sensitivity и specificity. Для оценки эффективности классификатора по разным классам используют метрики *специфичность* (specificity, или true negative rate, TNR) и *чувствительность* (sensitivity, или true positive rate, TPR):

$$\text{specificity} = \text{TNR} = \frac{TN}{TN + FP},$$

$$\text{sensitivity} = \text{TPR} = \frac{TP}{TP + FN}.$$

Таким образом, специфичность — это процент примеров негативного класса, которые были корректно распознаны, а чувствительность — часть корректно классифицированных позитивных примеров. В целом, чувствительность и специфичность можно рассматривать как метрику *accuracy* для образцов позитивного и негативного классов.

Данные метрики обычно используют в ситуациях ориентации на один класс, когда количество примеров, принадлежащих одному классу, существенно ниже чем общее количество примеров — в биоинформатике, обработке естественного языка, классификации текстов. Иначе говоря, в случаях, когда среди всех классов есть класс, представляющий особый интерес, а остальные классы либо объединены в один (бинарная классификация), либо оставлены как есть [37].

False positive и false negative rate. По аналогии с TPR в некоторых приложениях бывает полезно значение *FPR* (false positive rate), которое представляет собой отношение неправильно классифицированных отрицательных образцов к общему количеству отрицательных образцов. Другими словами, это доля отрицательных примеров, которые были неправильно классифицированы. Наряду с *FNR* (false negative rate), который показывает процент примеров положительного класса, которые были неправильно классифицированы, и дополняет чувствительность, *FPR* дополняет специфичность [38]:

$$\text{FPR} = 1 - \text{TNR} = \frac{FP}{FP + TN} = \frac{FP}{N},$$

$$\text{FNR} = 1 - \text{TPR} = \frac{FN}{FN + TP} = \frac{FN}{P}.$$

И *FPR*, и *FNR* не чувствительны к особенностям распределения данных, и поэтому могут использоваться в задачах с несбалансированными датасетами.

Precision и **recall**. Существуют метрики, которые позволяют оценить корректность классификации примеров для разных классов. Наиболее информативными среди них являются *точность* (precision) и *полнота* (recall), которая эквивалентна чувствительности:

$$precision = \frac{TP}{TP + FP},$$

$$recall = sensitivity = \frac{TP}{TP + FN}.$$

Является ли более важным precision или recall — зависит от специфики конкретной задачи классификации. К примеру, для задачи обнаружения болезни важным является способность классификатора корректно обнаружить как можно больше больных из общего набора, и в этом случае более важным параметром является recall. С другой стороны, важной характеристикой классификатора также можно назвать его чувствительность к неверной классификации, т.е. способность обнаруживать как можно больше примеров наличия заболевания, даже за счет ложно-положительных срабатываний — в таком случае важной характеристикой будет precision. Данные метрики, например, используются в работе [39], которая посвящена автоматической диагностике рака кожи по результатам дерматоскопии. Однако, в большинстве случаев приходится идти на компромиссы при выборе в качестве основной метрики между precision и recall.

Balanced accuracy. Иногда, в случае несбалансированных классов, используется *сбалансированная точность*, которая является средним чувствительности и специфичности и выражается формулой

$$BA = \frac{1}{2} \left\{ \frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right\}.$$

Если в задаче бинарной классификации количество примеров двух классов примерно поровну, то справедливо выражение $TP + FN \approx TN + FP \approx m/2$, где m — общее количество примеров, и сбалансированная точность примерно будет равна обычному значению ассурасу.

F-мера. В некоторых случаях бывает удобно каким-либо образом объединить точность и полноту в одно число, т.е. получить своеобразный агрегированный критерий точности работы алгоритма. В этом случае используют *F-меру*, которая является средним гармоническим точности и полноты вместо среднего

арифметического, что позволяет сглаживать расчёты за счет исключения экстремальных значений. В общем виде F-мера выглядит так:

$$F_{\beta} = (1 + \beta^2) \frac{precision * recall}{(\beta^2 * precision) + recall}.$$

Параметр β не что иное, как вес точности в метрике, и обычно в исследованиях используется значение $\beta = 1$, т.е. используют F_1 -меру:

$$F_1 = 2 * \frac{precision * recall}{precision + recall}.$$

F-мера достигает максимума при полноте и точности, равными единице, и близка к нулю, если один из аргументов близок к нулю.

Cohen's Kappa. Еще одним из коэффициентов, по которому можно судить о качестве работы классификатора в случае использования несбалансированных датасетов, является коэффициент *каппа Коэна* [40]. Основная идея данного коэффициента — перенормировка точности при помощи значения точности, которое можно было бы получить случайно. Выражается коэффициент формулой

$$k = \frac{accuracy - random\ accuracy}{1 - random\ accuracy}.$$

При этом *random accuracy* можно вычислить следующим образом:

$$random\ accuracy = \frac{(TP + FN)(TP + FP)}{FP + FN + TP + TN} \cdot \frac{(FN + TN)(FP + TN)}{FP + FN + TP + TN},$$

где первая часть означает вероятность того, что верно будет угадан класс P , а вторая часть — вероятность того, что верно будет угадан класс N . В таком случае, финальная формула для коэффициента k будет выглядеть так:

$$k = \frac{2 \cdot (TP \cdot TN - FN \cdot FP)}{(TP + FP)(FP + TN) + (TP + FN)(FN + TN)}.$$

Стоит отметить, что если значение метрики *accuracy* равно единице, то и значение коэффициента k будет равно 1. Этот коэффициент полезен в тех случаях, когда точность случайного угадывания достаточно высока (например, классы сильно разбалансированы, и предсказывание просто самого часто встречающегося класса даст высокий результат по сравнению с осмысленной классификацией).

Мультиклассовая классификация. Матрица ошибок может быть составлена не только для бинарного, но и для мультиклассового классификатора.

Пример такой матрицы ошибок для случая трех классов А, В и С представлен на рис. 29).

		Правильный класс		
		А	В	С
Предсказанный класс	А	TP_A	E_{BA}	E_{CA}
	В	E_{AB}	TP_B	E_{CB}
	С	E_{AC}	E_{BC}	TP_C

Рисунок 29 — Иллюстрация матрицы неточностей для мультиклассовой классификации, для случая трех классов

Здесь, по аналогии с матрицей для бинарного классификатора, зеленая диагональ — значения TP_A, TP_B, TP_C — обозначает количество верно предсказанных примеров для каждого из классов, оставшиеся ячейки матрицы — ошибочные предсказания. Например, E_{BA} — количество примеров, являющихся классом А, но предсказанных как В, E_{CA} — количество примеров, являющихся классом А, но предсказанных как С и т.д. Ложно-положительной (false positive) ошибкой для класса А будет сумма $FP_A = E_{BA} + E_{CA}$, т.е. количество примеров, которые были классифицированы как класс А, но им на самом деле не являются. Ложно-отрицательной (false negative) ошибкой для класса А будет сумма $FN_A = E_{AB} + E_{AC}$, которая показывает количество примеров класса А, ошибочно классифицированных как класс В или С. Таким образом, матрица неточностей размера $m \times m$ содержит m ячеек корректных классификация, и $m^2 - m$ ячеек с ошибочными результатами [38]. Характеристики FN, FP, TN, TP для всего классификатора вычисляются как сумма соответствующих характеристик по всем классам, т.е. $FN = FN_A + FN_B + FN_C$ и т.д.

В случае подсчета метрик для мультиклассового классификатора возможны различные варианты, не привязанных к конкретным классам:

- **микро**: вычисление глобальных метрик путем подсчета общего количества ложно-негативных (FN), ложно-позитивных (FP) и других

параметров (вместо подсчета индивидуальных метрик для каждого класса);

- **макро**: вычисление метрики для каждого класса и определение их невзвешенного среднего — в данном случае не учитывается сбалансированность датасета и вес каждой метрики среди всего набора;
- **взвешенное среднее**: вычисление метрики для каждого класса и определение их взвешенного среднего, опираясь на процентное соотношение количества образцов каждой метрики среди общего количества примеров в датасете — обычно используется в случае сильно несбалансированных датасетов.

Далее в тексте будет использоваться термин *среднеклассовые метрики*. В контексте данного исследования данная формулировка означает третий вариант подсчета — взвешенное среднее. При вычислении метрики для каждого конкретного класса при этом сам класс фиксируется как позитивный, а все остальные — как негативный (т.е. ситуация сводится к бинарному классификатору).

В последних работах, посвященных автоматической классификации остеоартрита, все чаще встречается использование коэффициента каппа Коэна (k) для оценки точности классификации, однако для большей наглядности и удобства сравнения с другими исследованиями в качестве основных метрик далее будут использоваться *precision*, *recall* и F_1 -мера. При этом, из-за особенностей подсчета параметров матрицы ошибок (TP — общее количество корректно классифицированных классов, $FN = FP$ — общее количество некорректно классифицированных классов) **recall** будет совпадать с **accuracy**. Более того, если бы в работе использовались микро-метрики, то можно было бы утверждать, что

$$micro-F1 = micro-precision = micro-recall = accuracy.$$

ROC-кривая и площадь под этой кривой

Одним из информативных методов визуализации качества бинарного классификатора считается кривая ошибок, или **ROC-кривая** (receiver operating characteristic). Изначально ROC-кривая использовалась (и продолжает использоваться) в теории обнаружения сигналов, для описания компромисса между

показателями угадывания и ложного срабатывания классификатора [41]. Одной из первых работ по машинному обучению, в которой упоминается ROC-кривая, считается [42], где кривая ошибок используется для оценки и сравнения алгоритмов. С развитием машинного обучения в целом и глубокого обучения в частности, упоминание ROC-кривой в литературе растет высокими темпами. Так, если в 1990 году было всего 50 упоминаний данного инструмента в научной литературе в год, то в 2002 это число достигло 1000, а в 2018 году зафиксировано более 8000 упоминаний кривой ошибок в год [43]. Стоит отметить, что ROC-кривая не зависит от распределения классов, что делает ее полезной для оценки классификаторов, прогнозирующих редкие события (болезни [44], стихийные бедствия и т.п.).

ROC-кривая — это двумерный график, на котором по оси Y откладываются значения TPR (или *recall*), а по оси X — значения FPR (или $1 - \text{specificity}$). График показывает относительный компромисс между выгодами (true positives) и затратами (false positives). Подробный алгоритм построения ROC-кривой можно найти в работе [41]. Рассмотрим рис. 30, на котором показаны четыре важные точки ROC-кривой — A, B, C и D.

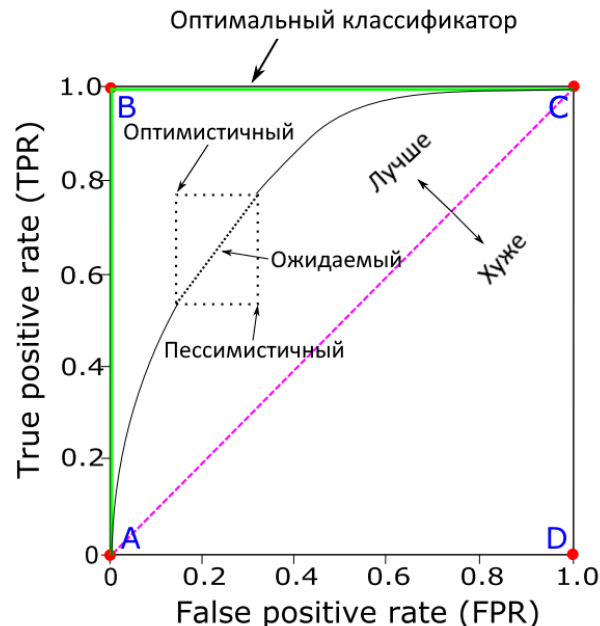


Рисунок 30 — Иллюстрация ROC-кривой с указанием важных для оценки классификатора точек, а также оптимистичный, пессимистичный и ожидаемый сегменты

Точка A с координатами (0, 0) представляет собой классификатор, который ошибочно классифицирует все позитивные образцы, но в то же время корректно

классифицирует все негативные ($TPR = FPR = 0$). Точка C с координатами $(1, 1)$, наоборот, представляет классификатор, который корректно классифицирует позитивный класс и все время ошибается на негативном классе.

Точка D в правом нижнем углу олицетворяет классификатор, который ошибается как на негативных, так и на позитивных примерах, а точка B — левый верхний угол — является идеальным классификатором, который корректно классифицирует и позитивные, и негативные образцы.

Помимо самой ROC-кривой, для сравнения нескольких классификаторов в различных исследованиях еще используют значение площади под этой кривой — **area under curve, AUC** [45]. Чем выше значение AUC — тем более качественней тестируемый классификатор, при этом значение 0,5 соответствует случайному угадыванию и означает, что классификатор непригоден для использования.

В дальнейшем для основных классификаторов будут представлены как матрицы ошибок, так и ROC-кривые вместе со значением AUC.

1.5 Обзор методов автоматической локализации коленного сустава

Существуют множество методов и подходов обнаружения объектов на изображениях, каждый из которых показывает хорошие результаты в определенных условиях или в случае работы с объектами, обладающими определенными признаками [46].

Одним из простейших методов локализации объектов является использование шаблонов (template matching), при котором для поиска объекта на изображении используется образец целевого объекта, сопоставляемый с некоторым набором положений этого объекта на изображении. Подобные методы различаются способами вычисления корреляции между шаблоном и изображением (т.е. может использоваться не только попиксельное сопоставление).

При локализации объектов, имеющих какие-то особые характеристики или признаки, эти признаки могут быть использованы для упрощения задачи. В случае с локализацией коленного сустава на рентгенограмме регион интереса также обладает определенными характерными визуальными особенностями, поэтому было предложено множество методов, которые эти особенности используют.

В работах [47; 48] авторы опираются на характерную форму костей сустава и расположение этих костей. Предлагаемый способ локализации областей коленного сустава состоит из двух этапов, первый из которых заключается в разделении левого и правого сустава на рентгенограмме, а второй является непосредственно локализацией сустава. Для этого используется вычисление центра масс гистограммы, посчитанной по сумме интенсивностей строк рентгенограммы (предполагая, что сустав расположен вертикально) после применения к ней фильтра Габора (рис. 31). Исследуются различные способы предобработки изображений для улучшения точности поиска центра масс — адаптивная эквализация гистограммы с ограничением контраста (contrast-limited adaptive histogram equalization, CLAHE [49]), применение фильтров Габора и др. В результате тестирования на наборе из 98 рентгенограмм (из датасета OAI) точность локализации при помощи вычисления центра масс гистограммы метод показал точность 84,38% для правого и 85,42% для левого суставов, а использование фильтра Габора в качестве предобработки перед этапом вычисления центра масс позволило повысить точность до 92,63% и 87,37% соответственно. К недостаткам метода можно отнести необходимость предварительной сегментации рентгенограмма на левый и правый сустав, низкое время работы и чувствительность метода к поворотам изображения.

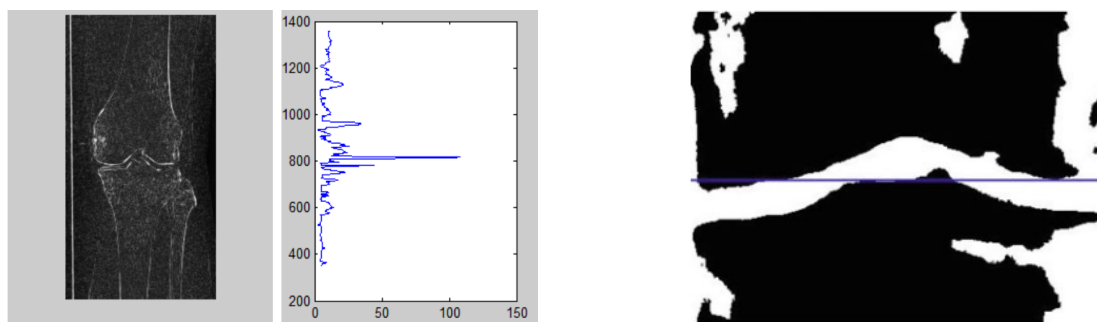


Рисунок 31 — Пример вычисления гистограммы по изображению, после применения к нему фильтра Габора [48] (слева) и пример области сустава, найденной при помощи гистограммы по бинарному изображению из работы [50]

Гистограмма суммы яркости по строкам также используется в [50], однако здесь гистограмма вычисляется по бинарной версии изображения. Сначала, как и в предыдущих работах, изображение подвергается процедуре эквализации гистограммы методом CLAHE, затем выполняется пороговое разделение преобразованного изображения, результат бинаризации инвертируется, чтобы пиксели хрящевой ткани межсуставного пространства стали белыми, и строится

гистограмма суммы пикселей по строкам. После нахождения центра гистограммы (строки с наибольшим количеством белых пикселей на бинарной версии изображения), в качестве области интереса выбирается область, находящиеся в пределах 60 строк по обе стороны от этого центра. Авторами отмечается, что в случае высокой степени ОА, при которой уже заметно сужено межсуставное пространство, метод может давать некорректный результат, а именно такие случаи, как правило, и представляют наибольший интерес при диагностировании.

Похожий способ с использованием бинарной версии для нахождения центральной части сустава представлен в [51]. Для повышения точности после бинаризации (с использованием детектора Кэнни [52] и морфологических операций) авторы используют совмещение бинарного изображения с несколькими эталонными шаблонами костей для отсеивания лишней информации, однако целевым объектом для поиска здесь является не область сустава, а его центр. В качестве данных для исследований использовались рентгенограммы 107 пациентов, из которых 4 были отобраны для составления эталонных шаблонов.

Метод из работы [53] также использует гистограмму интенсивностей по строкам. После вычисления гистограммы, для полученной последовательности значений считается одномерный градиент, по изменению которого фиксируются возможные кандидаты на принадлежность к центру сустава. После данной процедуры классификатор SVM, опираясь на гистограмму направленных градиентов (HOG) для каждого из выделенных регионов, выбирает из всех кандидатов лучшего (рис. 32). Для проверки работоспособности метода используется датасет MOST, состоящий из 1593 рентгенограмм, для обучения классификатора при этом используется 991 рентгенограмма, для валидации — 110, а для теста — 473. Точность локализации — среднее значение параметра IoU — для данного метода составила 83,99%.

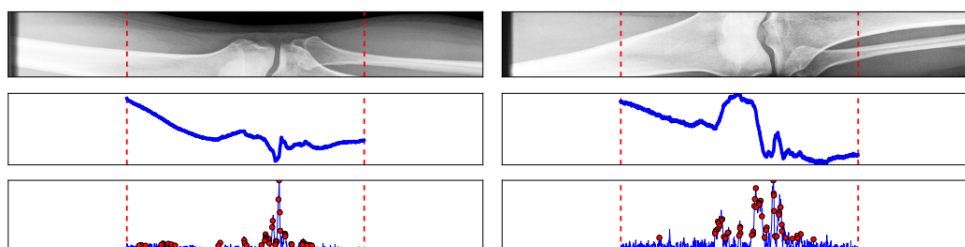


Рисунок 32 — Пример вычисления гистограммы по изображению, после применения к нему фильтра Габора [48] (слева) и пример области сустава, найденной при помощи гистограммы по бинарному изображению из работы [50]

Еще одними из базовых и относительно простых методов поиска специфических объектов на изображении являются каскадные классификаторы, которые основываются на методе скользящего окна с изменяемым размером. Например, подобный подход часто используют для решения задачи детектирования лиц на изображении — например, метод Виолы-Джонса [54].

Авторы работы [55] для локализации области коленного сустава используют сопоставление с шаблоном, в качестве которого выбрано типовое изображение сустава на рентгенограмме. При этом выполняется поиск области, которая лучше всего коррелирует с этим шаблоном. Похожий подход применяется и в работе [56] — в качестве шаблонов здесь используются 20 отобранных изображений области сустава размером 150×150 пикселей, центр которых совпадает с центром суставной щели. Во время поиска для избавления от деталей и ускорения исходное изображение, на котором необходимо найти область сустава, а также все шаблоны масштабируются до размера 10% от исходного (т. е. до размера 15×15 пикселей в случае шаблонов). Поиск выполняется методом скользящего окна, при этом для каждого положения вычисляется 20 евклидовых дистанций (по 1 на шаблон). В качестве центра сустава выбирается центр того окна, чье евклидово расстояние до шаблонов было наименьшим, а в дальнейшей обработке учувствует часть изображений размером 250×200 , чей центр совпадает с центром сустава. Данный метод обладает низкой точностью и очень чувствителен к шуму и отличию коленного сустава на рентгенограммах от базовых образцов.

В некоторых задачах широко используются методы, основанные на вычислении дескрипторов ключевых точек изображения, что позволяет довольно точно классифицировать и локализовать объекты на изображении. К таким дескрипторам можно отнести HoG [57], SIFT [58], SURF и др. Например, в работе [59] решается задача локализации ключевых точек коленного сустава по данным МРТ при помощи метода SIFT. Однако в случае локализации области сустава целевые объекты имеют слабую вариабельность (т.е. они, как правило, очень похожи друг на друга), что затрудняет использование подобных дескрипторов или делает их использование избыточным.

В [60] область сустава ищется при помощи скользящего окна и классификатора SVM, который по содержимому окна (после применения к нему горизонтального оператора Собеля для выделения горизонтальных краев на изображении) определяет, представляет ли область интерес или нет. В работе

используется скользящее окно размером 20×20 , которое символизирует центр сустава, а финальный центр берется как образец, на котором классификатор SVM дал наибольший отклик (рис. 33). После выбора финального центра, в качестве сустава для дальнейшей обработки берется область 300×300 вокруг этого центра. Перед процедурой левая и правая кости на снимке (если снимок содержит проекции обеих ног) должны быть разделены отдельно — т.е. метод предполагает, что на изображении возможно наличие лишь одного сустава. Использование автоматического классификатора позволило нивелировать недостатки простого метода — попиксельного сопоставления с шаблоном — связанные с возможными отличиями изображения от базовых образцов сустава, и получить точность локализации в 81,8%, по сравнению с 54,4% при сопоставлении с шаблоном. При этом, указанная точность означает лишь наличие какого-нибудь пересечения найденного ограничивающего прямоугольника с искомым ($J > 0$), если же поставить ограничение хотя бы 50% пересечения ($J > 0,5$), то точность сопоставления с шаблоном составляет всего 8,3%, а SVM — 38,6%.

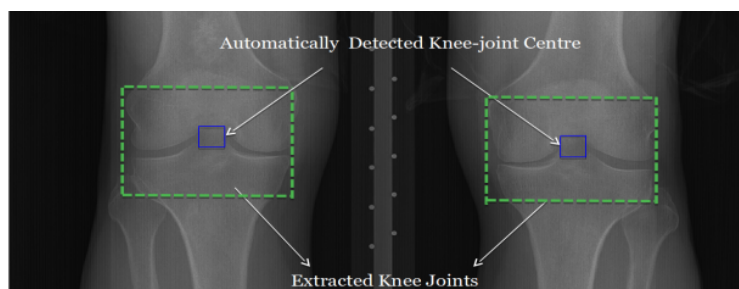


Рисунок 33 — Пример лучшего центра сустава (синий цвет) и финальной области сустава (зеленый цвет), выбранные после применения классификатора ко всем кандидатам из работы [60]

В [61] для локализации используется алгоритм т.н. локальных (region-based) активных контуров. Регион представлен набором точек, каждая из которых имеет определенную «энергию» (которая делится на внутреннюю и внешнюю энергии), и в процессе оптимизации каждая точка, независимо от других, стремится занять на изображении положение, в котором ее энергия максимальна. Авторы [62] для определения области сустава используют максимизацию некоторой функции, которая определяет опорную точку (reference point) области интереса, основываясь на наличии остеофитов на определенных частях костей.

После публикации AlexNet для локализации начинают использовать сверточные нейронные сети. Так, одной из первых архитектур, решающих эту задачу при помощи глубокого обучения, можно назвать архитектуру [63], которая основана на сети AlexNet с измененной функцией потерь.

Следующей важной разработкой можно назвать двухпроходную архитектуру RCNN. На первом этапе при помощи алгоритма Selective Search [64] составляется набор регионов-гипотез, в которых теоретически могут находиться искомые объекты. Selective Search основан на кластеризации содержимого изображения и последующим объединением полученных регионов по признакам однородности по цвету, текстуре, размерам и т.п. На втором этапе набор гипотез подается на вход сверточной нейронной сети, которая для каждого региона определяет вероятность наличия объекта того или иного класса и производит уточнение координат ограничивающих прямоугольников.

Улучшение RCNN было сделано в архитектуре Fast-RCNN и заключалось оно в том, что было предложено поменять местами применение сверточной сети для вычисления карты признаков изображения и алгоритм Selective Search. В такой конфигурации сверточную сеть больше не надо было применять к каждому региону, что позволило существенно ускорить процесс вычислений.

В работе [65] для решения задачи локализации области интереса авторы используют натренированную с нуля т. н. полностью сверточную сеть (Fully Convolutional Network, FCN). В отличие от традиционных сверточных сетей (CNN), у FCN отсутствуют полносвязные слои, а выходом в подобных сетях является изображение такого же размера, как и вход сети, каждый пиксель которого обозначает его принадлежность к тому или иному классу [66]. В противовес слоям пуллинга и сверточным слоям традиционных CNN, которые понижают размерность изображения, в FCN используются т.н. деконволюционные слои, которые повышают размерность изображения до требуемого размера. Архитектура сети представлена на рис. 34.

Точность локализации на датасете OAI для коэффициента Жаккарда $J > 0,5$ составила 99,9%, а для $J > 0,75$ — 89,3%, при этом среднее значение точности локализации достигает 0,83. При этом не требовалось отдельного разделение коленей на левое и правое на рентгенограмме, а можно было применять метод для всей рентгенограммы и получить все имеющиеся области расположения сустава на изображении. Точность локализации сустава при этом значительно выше

предыдущих предложенных методов. Тем не менее, существуют и более точные методы локализации.

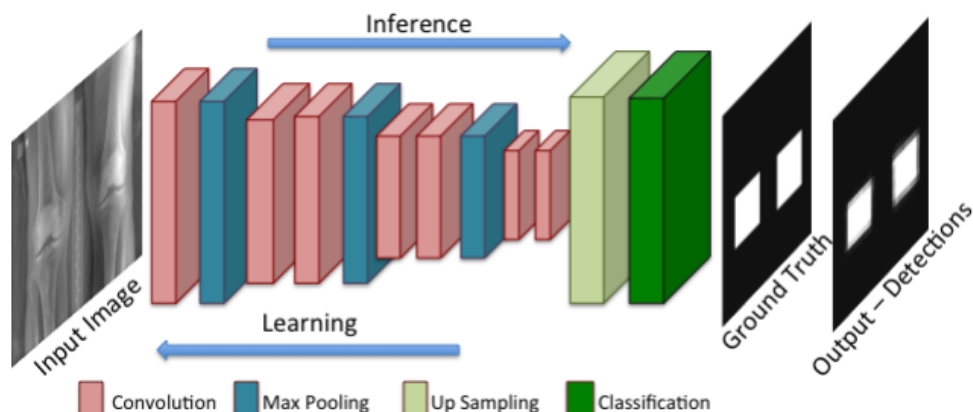


Рисунок 34 — Архитектура полностью сверточной нейронной сети для локализации коленного сустава из работы [65]

В работе [67] используется однопроходная архитектура YOLO, одна из первых однопроходных архитектур детекторов. Она отличается от предыдущих моделей тем, что за один проход сети позволяет получить набор предсказанных ограничивающих прямоугольников с наибольшей вероятностью наличия в них искомым объектов. Базовый набор возможных положений и размеров этих прямоугольников фиксирован и зависит от размера ячейки (виртуальной), на которые разбивается исходное изображение. После получения возможных положений объекта достаточно провести подавление не максимумов, и можно получить финальные положения и размеры объектов. Точность локализации составила примерно 82% на 200 аннотированных рентгенограммах.

В [68] для локализации области сустава используется дообучение сети с архитектурой YOLOv2. Модель накладывает на изображение сетку, а затем пытается предсказать вероятность наличия того или иного объекта (в виде комбинации ширины, высоты и центра ограничивающего прямоугольника) в каждой ячейке этой сетки, а также степень уверенности о наличии этого объекта. Поведение сети похоже на метод скользящего окна. Точность локализации на 1650 изображений из датасета OAI для $J > 0,5$ составила 99,9%, а для $J > 0,75$ — 92,2%, среднее значение — 0,858.

1.6 Обзор методов автоматической классификации остеоартрита коленного сустава

Существуют разные подходы к оценке остеоартрита, опирающиеся на разные типы данных. В качестве таких данных могут выступать медицинские рентгенограммы, данные МРТ, разного рода биомеханические и демографические данные и т.п. [69]. Например, в работе [70] приводится пример классификации степень ОА у пациентов, основываясь на таких данных, как возраст, рост, вес, пол, а также наличие определенных генов. В эксперименте участвовали 254 человека, и удалось достигнуть точности предсказания наличия ОА более 90%. Однако, без визуальной картины определить этап развития болезни все-таки довольно проблематично, поэтому большинство методов основаны на анализе графической информации.

В некоторых категориях (например, МРТ-исследования) в публичном доступе находится недостаточно данных, посвященных исследованию остеоартрита коленного сустава. Так, в работе [59], занимающейся исследованием снимков МРТ коленного сустава, в эксперименте используется лишь данные от 43-х пациентов. Это затрудняет применение высокоуровневых подходов вроде глубокого обучения. Поэтому, наиболее распространенными методами по автоматическому анализу являются методы, которые используют рентгенограммы.

Классические методы. Одним из показателей развития остеоартрита является изменение угла между большеберцовой и бедренной костями, т.е. косвенно развитие остеоартрита можно оценивать по значению этого угла. В работе [71] предлагается полуавтоматический метод, позволяющий определять этот угол по рентгенограмме. Метод состоит из таких этапов, как ручная расстановка опорных точек, которые определяют приблизительные контуры большеберцовой и бедренной костей, уточнении позиции этих точек при помощи метода активных форм, что в совокупности позволяет сегментировать регионы соответствующих костей. Затем выполняется скелетизация полученных регионов при помощи метода истончения, чтобы получить их центральные линии, на основе которых и выполняется определение угла между костями. Тестирование метода выполнялось на 50 изображениях из датасета OAI, при этом полученная ошибка измерения углов в среднем не превышала 0,81 градуса.

В [72] приводится обзор нескольких методов, основанных на использовании текстурных характеристик, которые используются для классификации ОА и локализации сустава, включая самоорганизующиеся карты Кохонена (self-organizing maps, SOM), фильтры Габора и модель активных форм (Active Shape Model) для определения геометрических характеристик костей.

Некоторые подходы основаны на низкоуровневой обработке изображений. Так, в [73; 74] просто автоматизированы обычные процедуры, которые выполняются экспертом при ручном измерении характеристик сустава. К таким процедурам можно отнести измерение сужения пространства между костями сустава, определение наличия остеофитов на боковых сторонах костей и измерение угла между бедренной и большеберцовой костью. Схожий с данным подходом метод, однако, являющийся полуавтоматическим (т. е. требует участия эксперта) и ограничивающийся измерением сужения сустава, описан в [75]. В [76] предлагается использовать алгоритм порогового разделения — в качестве порога выбирается средняя интенсивность на изображении, и затем измеряется пространство между суставом по бинарной маске. Автоматизация описанных выше действий требует определения контуров костей или границ их конкретных участков, что, как показано в [77], не лишено трудностей и предъявляет высокие требования к качеству рентгенограмм. Кроме того, подобные методы чувствительны к особенностям оборудования (что отражается на характеристиках рентгенограмм) и могут давать различные результаты на данных, полученных из разных источников.

Классические методы обработки изображений в последнее время все реже используются как отдельный полноценный инструмент решения задачи, и часто являются одним из вспомогательных шагов в составе более сложного алгоритма классификации.

Машинное обучение. Машинное обучение — это подраздел искусственного интеллекта, методы которого основаны на выявлении взаимосвязей и закономерностей в данных. С помощью статистических методов алгоритмы обучаются классифицировать и строить прогнозы на новых данных по зависимостям, которые удалось выявить из имеющихся данных. Отличительной способностью алгоритмов машинного обучения от алгоритмов глубокого обучения является то, что в первом случае набор признаков, по которым при помощи математических методов и вычисляются искомые зависимости, определяется вручную, тогда как в случае глубокого обучения этот набор признаков извлека-

ется автоматически в процессе обучения. Задача классификации является одной из типичных задач, решаемых методами машинного обучения.

Авторы [78] на практике оценивают применимость к классификации ОА таких методов, как деревья решений, наивный байесовский классификатор, байесовские (вероятностные) сети, логистическую регрессию и др. В тестировании использовался датасет из 130 изображений, при этом в классификации участвовали несколько различных участков сустава. Основой для классификации также, как и в описанных выше работах, являлись различные текстурные характеристики вроде текстурных признаков Харалика, особенностей гистограммы и др. В [79] предложен метод нормализации данных для вычисления текстурных характеристик, а также оценивается применимость для определения наличия ОА наивного байесовского классификатора и случайных лесов (без определения степени развития ОА).

Из универсальных методов классификации изображений по классам, не зависящих от постановки задачи или содержимого изображений, можно выделить WND-CHARM (the weighted neighbor distances using a compound hierarchy of algorithms representing morphology, [80]). Метод требует предварительно разметки некоторого набора тестовых изображений, которые являются типичными примерами каждого из классов, и основан на извлечении из изображения и его преобразований (Фурье, Чебышева, вейвлет-преобразований и т.п.) некоторого набора численных характеристик, которые затем используются для классификации и определения класса входного изображения. В оригинальной работе использовался вектор из 1025 характеристик, включающих пиксельную статистику (распределение интенсивностей, гистограммы, моменты), полиномиальную декомпозицию, высококонтрастные характеристики, такие как границы и объекты, и другие (рис. 35). Авторами отмечается, что высокоразмерный вектор характеристик на практике оказывается избыточным, поэтому предварительно производится оценка вклада каждой характеристики при помощи критерия Фишера (Fisher score [81]) и отсечение характеристик с наименьшим весом по 35%-му порогу (порог выбран эмпирически). Класс входного изображения определяется путем сравнения его вектора характеристик с каждым вектором из тестового набора при помощи модифицированного метода ближайших соседей, в котором дистанции между различными характеристиками имеют различный вес (weighted neighbor distances, WND). Метод был опробован на решении нескольких различных задачах классификации биологических

изображений и показал результаты, сопоставимые со специализированными для этих задач подходами, при этом сам метод не требовал никакой специальной настройки.

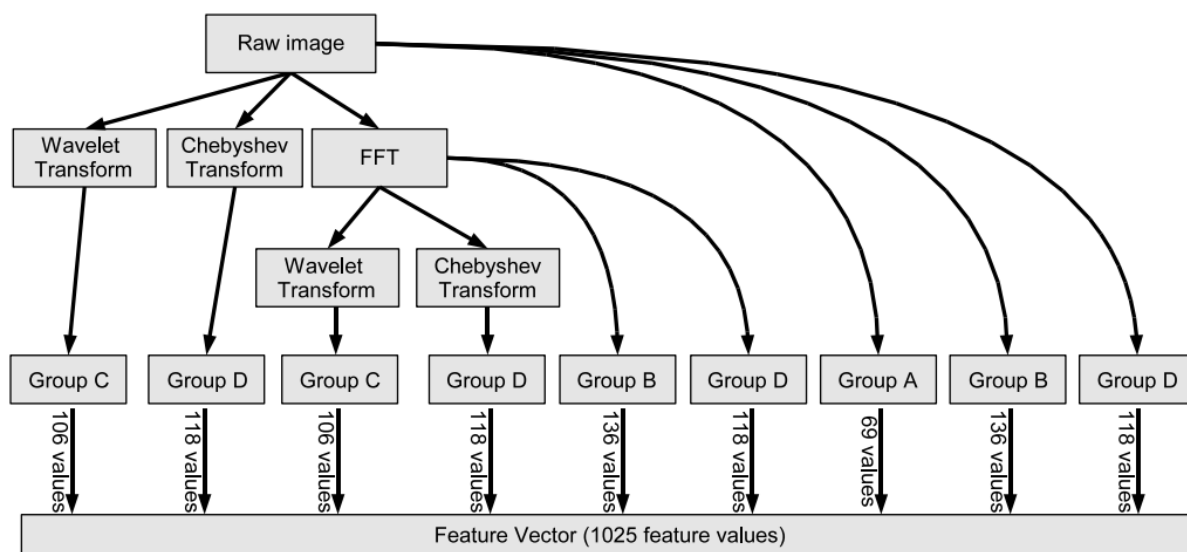


Рисунок 35 — Схема вычисления множества признаков из исходного изображения в методе WND-CHARM из оригинальной работы, многие признаки вычисляются повторно над различными преобразованиями исходного изображения — картой границ, вейвлет-преобразованием и др.

В работах [56], [82] приводится пример применения метода WND-CHARM непосредственно к задаче выявления ОА и классификации стадии развития ОА. В обоих случаях для тренировки классификатора и валидации было использовано в общей сумме порядка 150 изображений, при этом точность классификации различна для различных стадий ОА и колеблется в районе 38%. Далее будет показан пример применения метод WND-CHARM на имеющемся наборе данных из нескольких тысяч изображений различных стадий ОА.

В [83] предлагается методом случайных лесов анализировать характеристики не всего изображения, а непосредственно области сустава (участка между большеберцовой и бедренной костью). В данной работе, в отличие от [80], используется не такой большой набор характеристик. Это первые четыре момента, статистические характеристики (среднее значение интенсивности, значение энтропии), особенности формы (количество пикселей, периметр и т.п.) и текстурные признаки Харалика. В работе использовался набор из 200 изображений, из которых для обучения лесов были использованы 60%, а оценка происходила на оставшихся 40%. Приводится пример результатов тренировки как на отдель-

ных наборах характеристик, так и на общем наборе, при этом максимальная заявленная точность (при участии всех характеристик) равна 87,5%.

В работе [84] авторы работают с такой же областью сустава, как и в [83], однако в этом случае при помощи классификатора SVM принадлежность сустава тому или иному классу по шкале Келлгрена-Лоуренса определяется на базе гистограммы ориентированных градиентов. В эксперименте участвовало 616 изображений, размеченных двумя различными экспертами. Достигнутая точность классификации — более 90%.

В [85] предлагается оценивать степень прогрессирования ОА по рентгенограмме бедра, а именно по определенному участку текстуры в области примыкания кости бедра к тазу. Сама классификация выполняется при помощи комбинации нескольких методов, таких как байесовский классификатор, метод к-ближайших соседей и вероятностной нейронной сети (Probabilistic Neural Network, PNN). При этом используется шкала Келлгрена-Лоуренса от 0 до 4 (т.е. сокращенный вариант, в котором стадии 0 и 1 объединены в один класс), а в самом эксперименте участвует порядка полутора сотен рентгенограмм.

Другие подходы так или иначе основаны на использовании какого-либо классификатора. В [86], [50] авторы предлагают анализировать текстуру и отношения между пикселями (подсчет яркостных характеристик вроде стандартного отклонения, асимметрии и энергии, подсчет пар пикселей со специфическими значениями и т.п.) в области сустава, используя метод опорных векторов [50] или его комбинацию [50] с методом ближайших соседей. Заявленные результаты впечатляющие (80-100% точности), однако низкое число тестовых изображений (40 изображений в первом случае и 31 во втором) не позволяет в полной мере оценить качественные характеристики предложенного метода.

В [87] представлен оригинальный подход анализа рентгенограмм на наличие ОА по боковой проекции сустава. Оригинальность заключается в том, что в подавляющем большинстве работ для анализа рентгенограмм коленного сустава используется только фронтальная проекция. В своей работе авторы используют т.н. статистическую модель формы (Statistical Shape Model) для определения ключевых точек контуров костей сустава и случайные леса для начальной инициализации положения этих ключевых точек перед основным процессом фиттинга этих точек в более подходящие позиции. Затем полученная конфигурация точек используется для классификации стадии ОА методом случайных лесов. Метод был опробован на 300 рентгенограммах коленного су-

става в боковой проекции из датасета MOST (60 изображений на класс), при этом рассматривались две задачи — простое определение наличия признаков остеоартрита (стадии 0–1 — пациент здоров, стадии 3–5 — есть ОА) и непосредственно классификация стадии ОА по шкале Келлгрена-Лоуренса. В первом случае удалось достигнуть точности 78,9%, во втором — 47,9%.

Похожий способ классификации — случайные леса — используется в [88], однако здесь используется фронтальная проекция сустава, а помимо точек контура для классификации еще используется участок текстуры большеберцовой кости. Для тренировки классификатора и тестирования использовались 500 изображений из датасета OAI, при этом распределение изображений по степеням в этом наборе следующее: KL0 — 110 (22%), KL1 — 142 (28.2%), KL2 — 87 (17.4%), KL3 — 118 (23.6%), KL4 — 43 (8.6%). В результате экспериментов авторами была получена точность в 84%, однако в работе решалась задача простой бинарной классификации вместо классификации на 5 различных классов — т. е. лишь предсказывали, есть ли на представленном изображении признаки остеоартрита, или нет.

В последние годы популярными становятся подходы, использующие в своей основе различного рода нейронные сети [89]. В качестве исходных данных для этих сетей могут использоваться как рентгенограммы, так и данные МРТ, и данные ультразвуковых исследований [90]. Однако, из-за недостаточных объемов имеющихся в открытом доступе данных МРТ и ультразвуковых исследований, как правило, для автоматической оценки все-таки в большинстве случаев используют рентгенограммы.

В [62] для определения степени ОА используется простейшая нейронная сеть из двух слоев и двух нейронов, получающая на вход расстояние между костями сустава (joint space width). В [91] описывается комбинированный подход — сначала из изображения извлекаются различные характеристики, подобные таковым в методе WND-CHARM и т.п., а затем эти характеристики подаются на вход простой нейронной сети, состоящей всего из трех слоев (входной, выходной и один скрытый слой). Из-за маленького количества тренировочных изображений (менее сотни), обучения сети с нуля и простой архитектуры, на тестовой выборке авторам удалось достигнуть точности в 66,6%, что говорит о потенциальной применимости метода для решения поставленной задачи.

Однако, комбинированные подходы, использующие нейронные сети лишь в качестве одного из этапов классификации, обычно являются избыточными и

менее точными, чем полноценное использование глубокой нейронной сети (сети с большим количеством скрытых слоев) с конкретной архитектурой, поэтому в последнее время они практически больше не используются.

Глубокое обучение. Полноценным использование нейронных сетей можно назвать ситуацию, когда на вход сети подаются некоторые исходные данные, а сеть сама заботится об извлечении из этих данных необходимых для решения поставленной задачи характеристик в процессе тренировки [92]. Т. е. в отличие от подходов, подобных WND-CHARM, в этом случае нет необходимости вручную выделять характеристики изображения, которые будут использоваться для классификации — эти характеристики будут автоматически выявлены сетью в процессе обучения. Однако, для обучения нейронной сети, как правило, нужно большое количество данных, не всегда доступных исследователю. В случае с рентгенограммами коленного сустава эта проблема также актуальна, т.к. в открытом доступе есть всего несколько датасетов, каждый из которых включает в себя всего несколько тысяч изображений, что для полноценного обучения нейронной сети очень мало (так, одна из первых сверточных нейронных сетей AlexNet, участвующая в конкурсе по классификации изображений ImageNet, состояла всего из 8 слоев и тренировалась на датасете из 15 миллионов изображений). Де-факто стандартом для исследований в области классификации ОА являются датасеты из Osteoarthritis Initiative (OAI) и Multicenter Osteoarthritis Study (MOST). Датасет OAI состоит из 4796 рентгенограмм (в общей сложности доступно 9557 различных изображений отдельных коленных суставов), датасет MOST содержит 3026 изображений (среди которых можно выделить 5840 изображений коленей), при этом к каждой рентгенограмме имеется информация о проведенных измерениях, включая данные о стадии ОА (в различных шкалах, в т.ч. и по шкале Келлгрена-Лоуренса), наличии остеофитов, сужении межсуставного пространства, возрасте пациента и т.п.

Одна из первых нейронных сетей, полноценно используемая для классификации ОА, описана в [65]. Эта сеть состоит из 6 слоев, 5 из которых сверточные, а один — полносвязный, и принимает на вход изображения коленного сустава размером 200×300 (рис. 36). Авторы исследуют различные формулировки задачи — т.е. рассматривают как проблему классификации ОА (т.е. разбиения на 5 классов), так и проблему регрессии — предсказания числа, наиболее близкого к степени ОА на соответствующем изображении. Финальная точность, которой удалось добиться при обучении сети одновременно для реше-

ния задачи и классификации, и регрессии, составляет 63,4%. Похожий результат в 64,6% получен теми же авторами в более поздней работе [92], в которой проведен более глубокий анализ используемого подхода и результатов.

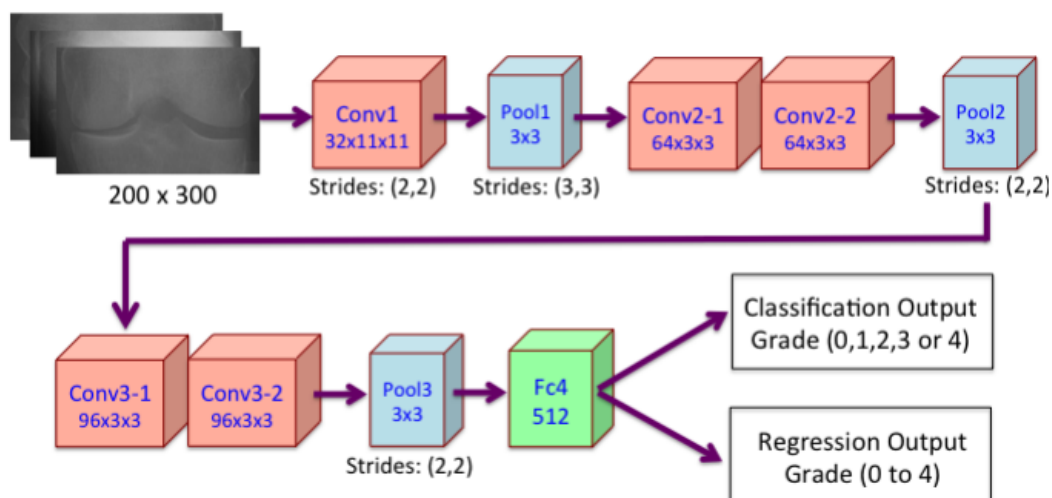


Рисунок 36 — Пример одной из первых архитектур нейронной сети из работы [65], используемой для классификации и регрессии стадии остеоартрита по шкале Келлгрена-Лоуренса

Несмотря на кажущееся обилие данных, для обучения с нуля нейронной сети этих данных может быть недостаточно, поэтому распространенным подходом является использование уже обученной на большом наборе данных нейросети и ее дообучение на нужном наборе данных. Для этого у обученной нейронной сети просто убирается последний слой, отвечающий за классификацию, добавляется другой, который требуется для решения поставленной задачи, и нейронная сеть с уже обученными весами снова подвергается обучению, но уже на другом, небольшом, наборе данных. В [60] показаны результаты использования для классификации стадии ОА предтренированных сверточных нейронных сетей с различными архитектурами, которые дообучаются на новом датасете с соответствующими примерами.

В [93] исследуется использование для классификации сверточных сетей с архитектурой, в основе которой лежит DenseNet. При этом рассматривается как классификация исключительно по изображению, так и дополнение последнего полносвязного слоя (после прохождения основных данных через сверточные слои) такими параметрами, как пол пациента, его возраст и раса. Для локализации коленного сустава на рентгенограмме в работе используется сеть с архитектурой U-Net, которая позволила достичь точности локализации 98,3% на выборке из 1000 изображений. Для оценки прогрессирования остеоартрита

используется шкала Келлгрена-Лоуренса, у которой 0-1 и 1-й классы объединены в один, означающий отсутствие остеоартрита. Среднеклассовая точность классификации на тестовом датасете в итоге составила 77,2%, при этом использование дополнительных демографических данных не принесло статистически значимого улучшения.

Работа [94] использует комбинацию нескольких схем нейронных сетей над изображениями коленного сустава размером 400×100 (выделенными для каждой рентгенограммы датасета OAI вручную). При этом использовался не полный датасет OAI, а его часть, состоящая из 1500 изображений (до проведения аугментации в виде случайного поворота в диапазоне $[-10, 10]$ градусов). Сначала для извлечения карт признаков, используется VGG-16, затем полученные карты признаков подаются на вход сети LSTM (Long Short Term Memory), которая по результатам вычислений и предоставляет выходной вектор распределения вероятностей принадлежности сустава на изображении к той или иной степени остеоартрита по шкале Келлгрена-Лоуренса. Обучение выполняется при помощи метода стохастического градиентного спуска (Stochastic Gradient Descent, SGD). В данной работе максимальная точность классификации на тестовом датасете из 450 изображений составила 75,28%

Авторы [95] предлагают использовать в традиционной схеме со сверточными слоями и слоями понижения размерности с архитектурой VGG-16 еще и обучаемые модули внимания (attention modules), а саму сеть разделить на несколько ветвей (в данной работе используется три ветви), каждая из которых отделяется от основной последовательности слоев на разных уровнях абстракции (рис. 37). Каждая ветвь при этом заканчивается вектором признаков, полученным при помощи глобального усредняющего пуллинга (GAP). Финальный ответ получается после прохождения агрегированных данных от всех трех ветвей через полносвязный слой и слой Softmax. При использовании данного подхода удалось достигнуть точности классификации в 64,3%.

В [96] используется комбинированная сверточная сеть архитектуры Faster R-CNN. Данная архитектура за один проход позволяет осуществить локализацию коленного сустава и выполнить процесс предсказания стадии остеоартрита без необходимости явного разделения процесса на стадию локализации и классификации. В качестве датасета использовался набор из 1385 изображений коленей. Для увеличения объема к данному датасету была применена техника аугментации в виде горизонтального отражения, что позволило удвоить коли-

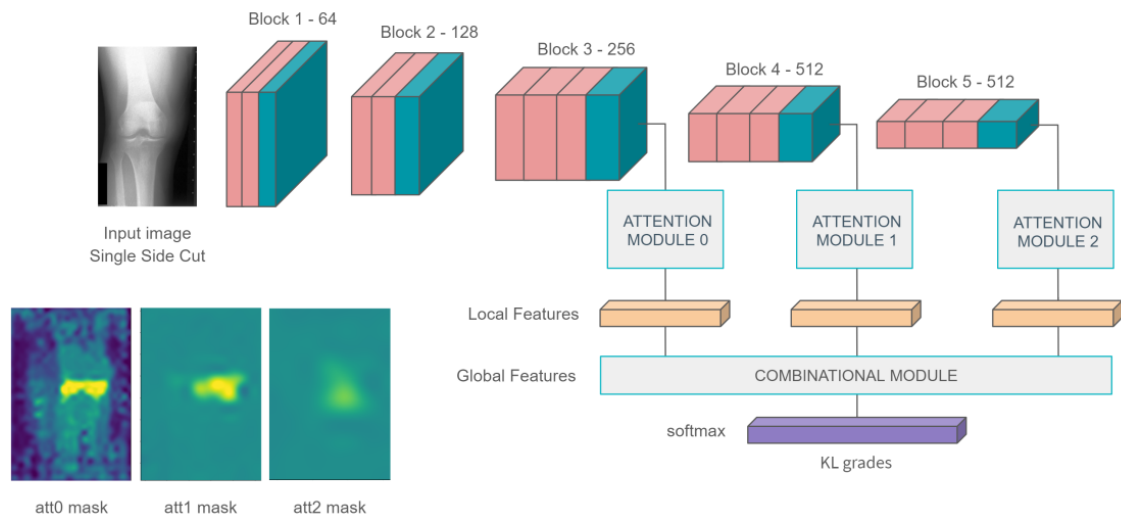


Рисунок 37 — Пример интеграции модулей внимания в архитектуру VGG-16 из работы [95]

чество примеров. Все изображения были затем вручную оценены экспертом и разделены на 5 стадий остеоартрита по шкале Келлгрена-Лоуренса. В качестве функции потерь в процессе обучения сети использовалась *focal loss*. Точность классификации на тестовом датасете из 550 изображений в данной работе равна 76,5%.

В работе [68] авторы предлагают использовать специальную функцию потерь (ordinal loss), которая по-разному штрафует несоответствия между истинными и предсказанными степенями ОА. Предлагаемая ими функция потерь позволяет сильнее штрафовать более серьезные несоответствия — к примеру, штраф за ошибку между 1 и 2 степенью ОА будет меньше, чем за ошибку между 1 и 4. В работе, с использованием предложенной функции потерь, были натренированы несколько сетей известных архитектур, такие как ResNet различной глубины, VGG и Inception-v3. Проведены сравнения между использованием стандартной функции потерь (кросс-энтропии) и разработанной. В результате авторам удалось добиться среднеклассовой точности в 67,7% на сети с архитектурой VGG-19, при использовании предложенной ими функции потерь.

Некоторые исследования используют особенности формы сустава для построения эффективных архитектур нейронных сетей. Так, в работе [97] берется в расчет симметричность коленного сустава на рентгенограмме относительно центральной оси. Предложенная архитектура является т.н. сиамской нейронной сетью, и состоит из двух частей, каждая из которых получает на вход свою часть изображения сустава — левую и правую соответственно, с перекрытием

(рис. 38). Каждая из подсетей выполняет вычисления независимо от другой и не делится своими весами. На выходе сети результаты работы каждой из подсетей объединяются и подаются на вход финальному слою классификации. Финальная среднеклассовая точность классификации составила 66,71%.

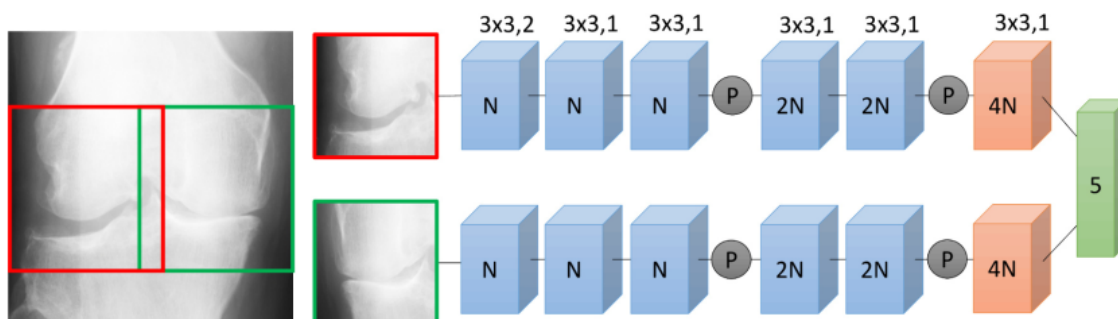


Рисунок 38 — Пример сиамской нейронной сети из работы [97]. Из имеющейся рентгенограммы выбираются два патча и подаются на вход различным веткам сети. Синие блоки на изображении — стандартная комбинация сверточного слоя, нормализации батчей (BN) и функции активации ReLU, серые круги — слои пуллинга, светло-красные блоки — блоки глобального усредняющего пуллинга, а зеленый блок — слой Softmax

В [98] рассматривается проблема локализация ключевых анатомических точек коленного сустава при помощи сверточной сети специального вида (Hourglass Convolutional Network). Конфигурацию этих точек затем можно использовать для классификации ОА.

В [99] авторы помимо предсказания ОА по шкале Келлгрена-Лоуренса обучают сеть для предсказания наличия остеофитов в суставе (в потенциальных участках их возникновения) и предсказания сужения межсуставного пространства (Joint Space Narrowing, JSN). Для этого используется комбинация из двух глубоких нейронных сетей семейства ResNet (Resnet-18, Resnet-34 и Resnet-50), результаты предсказания которых усредняются для получения финального результата. Результатом является степень уверенности в наличии остеофитов в 4 частях изображения, на краях костей (рис. 39), сужение межсуставной щели и общая классификация ОА по шкале Келлгрена-Лоуренса. При этом, для тренировки сети используется перенос обучения с ImageNet. Для предсказания степени остеоартрита по шкале Келлгрена-Лоуренса сбалансированная точность (т.е. точность, полученная как взвешенная сумма точностей по каждой градации шкалы KL) составила 66,68%.

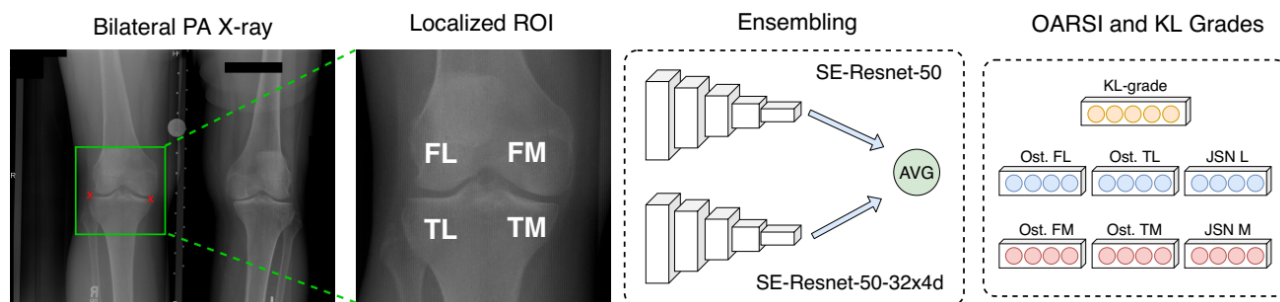


Рисунок 39 — Общая схема метода из работы [99]. Сеть предсказывает степень ОА по атласу OARSI для бедренной боковой (FL), большеберцовая боковой (TL), бедренной медианной (FM) и большеберцовый медианной частей кости (TM), а также сужение межсуставного пространства для бокового (JSN L) и медианного (JSN M) отделов сустава

В работе [100] решается не просто задача классификация степени ОА, а задача определения прогрессирования ОА, для решения которой используется не только рентгенограммы, но также и антропометрические данные (возраст, вес, индекс массы тела пациента), клиническая история, оценки эксперта (опционально) и др. В качестве базовой сети для извлечения признаков используется SE-ResNeXt-50, для объединения выхода нейронной сети с антропометрическими данными применяется градиентный бустинг [101] или логистическая регрессия. Все эти данные используются для предсказания вероятности прогрессирования ОА с течением времени. Максимальная точность, достигнутая при использовании всей доступной информации (включая предполагаемые оценки эксперта о возможной степени ОА), составила 70%, использование только данных с выхода сверточной сети позволила достичь точности 68%.

Комбинированная задача решается в [102]. Помимо классификации остеоартрита при помощи сверточной сети VGG, при помощи сети с архитектурой U-Net решается задача сегментации костей сустава и выделения контуров этих костей, на основе которых происходит вычисление сужения межсуставного пространства для бокового и медианного отделов сустава (JSN). Затем результаты классификации, полученные при помощи VGG-16, объединяются с парой значений JSN, и финальный результат классификации вычисляется при помощи случайного леса. На датасете MOST точность классификации одной VGG-16 в работе составила 71,93%, при помощи случайного леса по набору — 75,86%. На датасете OAI точность комбинированного метода с использованием случайного леса составила 64,48%, сбалансированная точность на этом датасете — 68,56%.

Оригинальная идея представлена в работе [103], где помимо рентгенограмм коленного сустава предлагается использовать некоторые характеристики походки пациента (частота шагов, ширина шага, угол сгибания колена, угол разгибания бедра и др.). Датасет состоит из 728 рентгенограмм коленей 364 пациентов с прикрепленной информацией о походке. В качестве метода классификации используется классификатор SVM, который опирается на особенности походки и карту признаков рентгенограммы, извлеченную при помощи нейронной сети с архитектурой Inception-ResNet-v2 [28]. Полученная среднеклассовая точность классификации данного метода — 64,7%.

1.7 Обзор методов выделения контуров на изображении

Существует большое количество различных подходов к выделению контуров на медицинских изображениях. К таким подходам можно отнести сегментацию, деформируемые модели, а также методы, использующие имеющуюся в случае обработки медицинских изображений априорную информацию о целевых объектах. Также существуют различные интерактивные техники выделения контуров, которые могут состоять из комбинации вышеописанных алгоритмов, и на финальном этапе предлагают пользователю самостоятельно оценить результат детектирования и вручную откорректировать контуры в случае необходимости [104].

Сегментация и выделение контуров — две взаимосвязанные задачи. Если у нас есть в наличии размеченная область изображения, мы легко можем получить ее контур, и наоборот. Однако, методы сегментации опираются на предположение, что объект (или группа объектов) имеет некую однородную структуру, которая может быть формализована при помощи разного рода критериев однородности пикселей. Методы, занимающиеся выделением контуров на изображении, более гибкие и могут использоваться для выделения объектов, имеющих различные характеристики текстуры.

К классическим методам сегментации можно отнести пороговое разделение, алгоритмы, использующие карту границ изображения (edge-based) и алгоритмы, основанные на регионах (region-based). Пороговое разделение является простейшей техникой сегментации, но может быть использовано лишь

в случае бимодальной гистограммы изображения, т. е. тогда, когда содержимое этого изображения может быть разделено на объект и фон при помощи сравнения интенсивности пикселей с некоторым порогом. Существуют как глобальные алгоритмы порогового разделения, которые применяют порог сразу ко всему изображению, так и адаптивные, которые разбивают изображение на множество регионов, внутри которых можно однозначно отделить фон от объекта, и затем используют для них разные значения порогов (например, зависящие от содержания каждого региона). В [105] приводится пример использования адаптивного алгоритма порогового разделения для сегментации участка мозга, который учитывает анатомические особенности человека.

Довольно широкое применение в обработке медицинских изображений также находят методы, основанные на выделении регионов путем итеративной группировки пикселей — метод разрастания регионов [106] и алгоритм водораздела [107]. Такие методы обычно используются для сегментации медицинских изображений, в которых целевые объекты имеют однородную текстуру (например, сегментация опухолей или мозга на МРТ-снимках).

Однако, в случае использования медицинских рентгенограмм, на которых объектами интереса являются кости, на изображении могут присутствовать участки, которые не позволяют однозначно отделить объект от фона и не имеют четкого критерия однородности пикселей (рис. 40). Это затрудняет использование упомянутых алгоритмов.

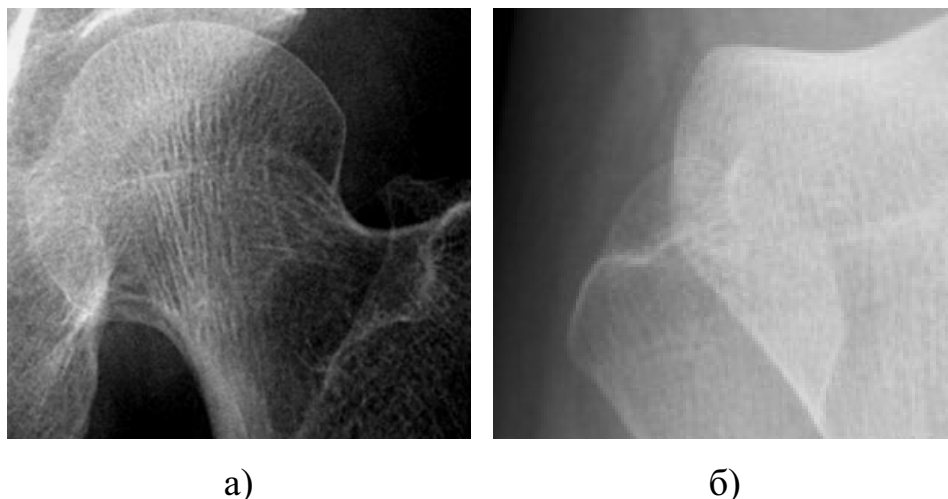


Рисунок 40 — Крупный вид участка бедренного сустава с неоднородной структурой (а) и участок с перекрытием двух костей (б)

Известным алгоритмом, используемым для выделения контуров и сегментации, является детектор границ Кэнни [52]. В частных случаях этот

детектор иногда позволяет получить хорошие результаты, однако требует ручного подбора параметров для каждого отдельного изображения. Также существуют специализированные алгоритмы отслеживания контуров. Основной их идеей является поиск некоторых особых точек объекта (углов, локальных максимумов яркости и др.) в качестве начальной и последующее отслеживание контура по другим особым точкам, учитывая только сильные (по условиям алгоритм) границы. Примером такого метода является алгоритм, предложенный в работе [108] для извлечения контуров из цветных изображений.

Подобные алгоритмы также не лишены недостатков, и требуют наличия четких границ и хорошей отделенности объекта от фона. Выше отмечалось, что особенностями рентгенографических изображений является большая нерегулярность текстуры и наличие сильного шума (рис. 40, а). Кроме того, частым случаем является наличие на рентгенограмме перекрывающихся проекций соседних костей, когда на изображении на одном и том же участке присутствуют границы нескольких объектов, трудноотделимые друг от друга (рис. 40, б). Эти проблемы не позволяют использовать известные алгоритмы отслеживания контуров с приемлемыми результатами.

Деформируемые модели — это подход, который основан на использовании шаблона целевого объекта, заданного контуром или их набором, который деформируется под действием различных сил и под их действием стремится принять форму объекта. Здесь выделяют методы активных контуров и модели активных форм.

Модель активных форм представляет из себя статистическую модель, сгенерированную из набора тренировочных изображений. Для ее построения на тренировочных изображениях фиксируются опорные точки, определяющие границу исходного объекта [51; 109], и затем из этих тренировочных изображений строятся наборы распределений различных параметров конфигурации точек, на основе которых и происходит идентификация объекта на изображении и т.п. В работе [51], например, финальный набор точек используется для вычисления сужения межсуставной щели.

Алгоритм активных контуров — это итеративный алгоритм, основанный на минимизации энергии исходного контура путем его деформации [110]. Энергию контура при этом обычно делят на части — внутреннюю (которая ограничивает эластичность и изгиб контура), внешнюю (которая ограничивает деформацию контура) и энергию изображения, которая учитывает такие факторы, как цвет

или границы объектов. Пример использования деформируемых моделей к сегментации костей кисти представлен на рис. 41 [111].

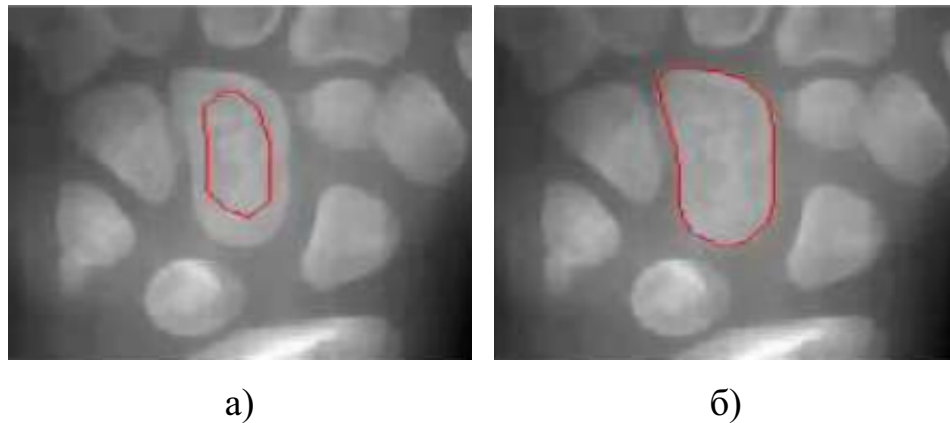


Рисунок 41 — Пример сегментации костей кисти руки из работы [111]: начальный контур (а) и результат применения алгоритма активных контуров (б)

В основном, методы активных контуров очень чувствительны к шуму и требуют хорошей инициализации. В качестве решения проблемы инициализации часто применяются алгоритмы, которые для этой цели используют различные априорные знания об объектах. В качестве априорных знаний может быть использована информация о связи между объектами на изображении [112], особенности формы целевых объектов [51; 113] и т.п. В работе [113], посвященной вопросам выделения контуров бедренной кости, предлагается для определения начальной позиции шаблона использовать поиск конфигурации параллельных прямых и кругов, характерной для объекта исследования (рис. 42, а). Тем не менее, даже подобные алгоритмы зачастую не могут справиться с сильным изменением формы объектов, что в случае медицинских рентгенографических изображений и представляет особый интерес (рис. 42, б-в).

Еще одним из современных подходов к детектированию контуров и сегментации является использование сверточных нейронных сетей и глубокого обучения. Одним из недостатков подобных методов является необходимость разметки больших датасетов с изображениями целевых объектов для обучения — причем для каждого объекта в датасете должен содержаться его точный контур. Готовых датасетов, пригодных для задач детектирования контуров, в публичном доступе не так много, и большинство наборов в области медицины заточены под использование в области КТ и МРТ, а не рентгенографии. Несмотря на то, что использование современных инструментов позволяет ускорить процесс

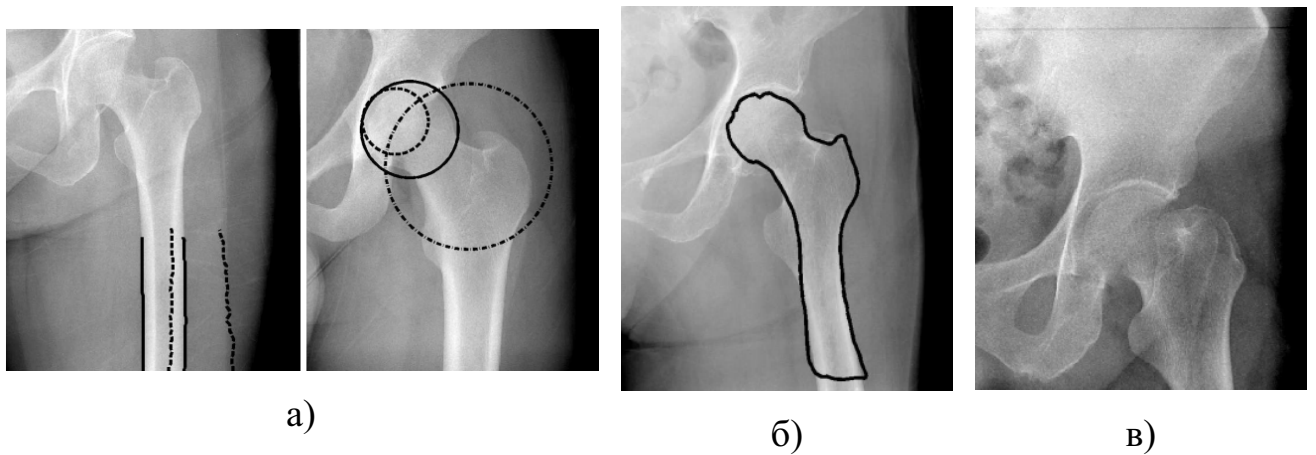


Рисунок 42 — Примеры элементов для поиска начальной позиции контуров (а), неточного описания шаблоном формы реального объекта (б) и пример нестандартной формы бедренного сустава, что не позволило его идентифицировать при помощи имеющихся шаблонов (в)

разметки, для каждой новой задачи необходимо провести большой объем предварительной работы.

1.8 Выводы

В главе приведен обзор основных существующих медицинских методов оценки остеоартрита коленного сустава, включая как количественные (измерение сужения и т.п.) методы, так и комбинированные (различные шкалы, атласы). Представлено описание существующих открытых наборов данных, пригодных для проведения исследований в области диагностики ОА.

Также в главе представлен обзор популярных и широко используемых архитектур сверточных нейронных сетей в хронологическом порядке. Для каждой из основных архитектур помимо подробного описания ключевых инженерных решений приводится графическая схема и подробная таблица с информацией о параметрах всех используемых в этих архитектурах слоев. Данная глава показывает, что существует большое количество архитектур со своими особенностями и преимуществами, которые могут быть использованы для решения задачи классификации остеоартрита коленного сустава по рентгенограмме.

Также помимо описания методов оценки остеоартрита и наборов данных, приведена информация о распространенных способах оценки качества работы

классификации и локализации, которые будут использоваться в рамках исследования.

Таким образом, в первой главе:

- описаны используемые для исследований наборы данных;
- описаны существующие подходы к оценке прогрессирования остеоартрита коленного сустава, особенности этих подходов и определена шкала оценки ОА, которая будет использоваться в дальнейшем;
- выполнена разметка датасета OAI, направленная на выделение областей сустава, представляющих интерес при решении задачи классификации остеоартрита;
- проведен подробный обзор современных архитектур сверточных нейронных сетей, включая строение моделей и архитектурные решения, позволяющие повысить точность классификаторов;
- представлено описание и разбор методов оценки точности классификации и локализации, которые будут использованы в рамках исследования;
- выполнен анализ полученных результатов.

Некоторые результаты из данной главы опубликованы в работе [139].

Глава 2. Разработка и исследование метода автоматической локализации коленного сустава

Данная глава посвящена исследованию методов локализации области коленного сустава на изображении.

Процесс определения степени остеоартрита по рентгенограмме сустава субъективен и сильно зависит от квалификации и опыта эксперта. Использование автоматизированного анализа могло бы снизить влияние фактора субъективности и повысить надежность и точность диагноза.

Определение наличия остеоартрита коленного сустава у пациента возможно по косвенным признакам, таким как искривления костей, изменения угла между костями и т.п. Тем не менее, наибольший интерес при исследовании остеоартрита представляет область сустава, т.к. именно там, как правило, сосредоточены основные признаки ОА. К таким признакам относятся сужение межсуставной щели из-за износа хрящевой ткани и деформация формы костей в этой области. Кроме того, важным критерием степени ОА считается наличие остеофитов по краям двух сочленяющихся костей. Остеофиты — это патологические наросты на кости, которые возникают при сильном износе хряща и неравномерного распределения нагрузки на поверхность костей. Именно остеофиты являются одной из основных причин болей при возникновении остеоартрита и фактором, ограничивающим движение сустава.

Таким образом, большинство автоматических методов, работающих с рентгенографическими изображениями, фокусируются именно на области сустава для определения степени прогрессирования остеоартрита. Именно поэтому для повышения точности классификации и акцентировании классификатора непосредственно на области интереса, необходимо выполнить локализацию этой области интереса (или нескольких областей) на всей рентгенограмме. В главе описывается итоговый метод локализации, реализованный и используемый в диссертации [147], а также приводится описание составляющих его компонентов в виде сверточных нейронных сетей.

2.1 Описание метода

Архитектуры для детектирования объектов на изображении при помощи нейронных сетей можно разделить на несколько типов, где одними из основных являются так называемые двухэтапные (two-stage) и одноэтапные (one-stage) подходы. В первом случае к выходу сети дополнительно применяется алгоритм генерации ограничивающих прямоугольников, который и позволяет получить положение объектов на изображении. К таким сетям относятся архитектуры R-CNN, Fast R-CNN, Faster R-CNN и др [114]. Одноэтапные модели сразу, без дополнительных алгоритмов, предсказывают положения ограничивающих прямоугольников объектов, опираясь на результирующую карту признаков. Такими архитектурами являются, например, YOLO [115] и SSD [116].

Применимость одной из модификаций архитектуры YOLO для локализации коленного сустава была исследована в [68], в диссертации предлагается решать эту задачу при помощи использования сверточной нейронной сети с архитектурой SSD (Single Shot Detector). При этом сам детектор SSD используется только для локализации коленного сустава, а классификацией занимается отдельная сверточная нейронная сеть.

2.1.1 Архитектура Single Shot Detector

Архитектура SSD является сетью прямого распространения (feed-forward), которая на выходе возвращает фиксированный набор ограничивающих прямоугольников и оценку вероятности наличия в этих прямоугольниках объектов того или иного класса, в результате постобработки которого можно получить финальный результат детектирования. Сама сеть при этом состоит из трех основных уровней.

Первый уровень. Первый (базовый) уровень состоит из предобученной сети какой-нибудь стандартной архитектуры (VGG-16, Resnet-101, Inception [26] различных версий и др [117]). На данном уровне от стандартной архитектуры используются первые несколько слоев, которые отвечают за извлечение низкоуровневых признаков из входного изображения. В работе для извлече-

ния признаков была использована предобученная на датасете COCO (Common Objects in Context) от Microsoft сеть с архитектурой MobileNet [30] — данная архитектура позволяет достигать точности, сравнимой с архитектурой VGG-16, обладая при этом количеством весов, на порядок меньшим, нежели в VGG-16, и достигая высокой скорости работы (в оригинальной работе 80% работы детектора занимало именно вычисление карты признаков сетью VGG-16 [116]).

Второй уровень. Второй (вспомогательный) уровень состоит из четырех блоков сверток, каждый из которых понижает пространственную размерность входных данных, что позволяет искать объекты различного масштаба. Это возможно благодаря тому, что выход каждого из таких блоков затем используется для локализации. Каждый блок, в свою очередь, состоит из пары сверточных слоев, первый из которых сначала понижает канальную размерность данных, а второй эту размерность снова повышает, но при этом уменьшает размерность пространственную.

Третий уровень. Третий уровень предназначен для вычисления фиксированного множества предсказаний координат центров прямоугольников, вероятности наличия объекта того или иного класса и т.п. при помощи сверточных слоев для выхода каждого из блоков вспомогательного уровня и — опционально — выхода некоторых слоев из базового уровня.

Каждой ячейке карты признаков соответствует набор ограничивающих прямоугольников с одинаковым и фиксированным положением и различным масштабом и соотношением сторон (рис. 43). Масштаб прямоугольников для каждой карты признаков вычисляется по формуле

$$s_k = s_{min} + \frac{s_{max} - s_{min}}{m - 1}(k - 1), k \in [1, m],$$

где m — количество карт признаков, используемых для локализации объектов, $s_{min} = 0,2$, $s_{max} = 0,9$. Карта признаков с номером $k = 1$ является самой низкоуровневой, $k = m$ — самой высокоуровневой, и это означает, что у первой карты признаков коэффициент масштабирования равен 0,2, а у последней (самой высокоуровневой) — 0,9. Также для каждой карты признаков используется собственный набор соотношений сторон a_r из множества $\{1, 2, 3, 1/2, 1/3\}$

На выходе детектора каждый такой прямоугольник представлен набором $(cx, cy, w, h, c_1, \dots, c_p)$, где (cx, cy) — координаты смещения центра прямоугольника относительно базового положения, (w, h) — модификатор для базового

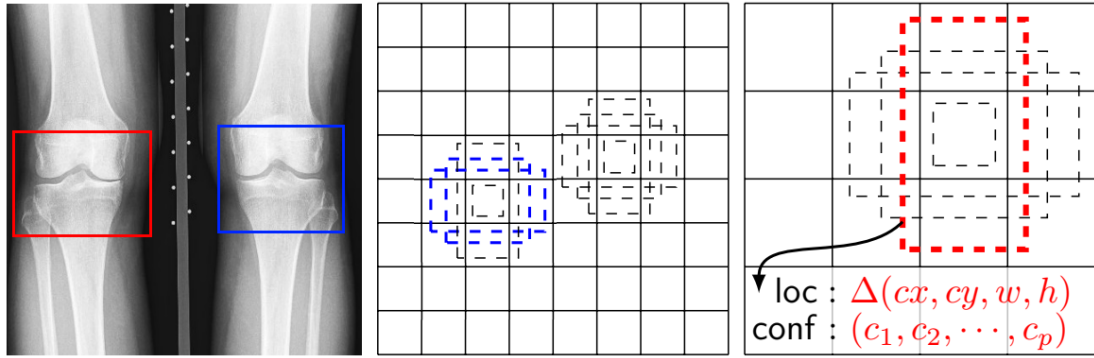


Рисунок 43 — Иллюстрация выбора положений для исследования наличия объектов на карте признаков. Слева — искомые ограничивающие прямоугольники, в середине — разбиение карты признаков сеткой 8x8, справа — разбиение сеткой 4x4 и параметры искомого прямоугольника

размера прямоугольника, а $(c_1, \dots, c_p)$ — вероятности наличия в этом прямоугольнике объекта соответствующего класса (для случая p классов).

Для каждой карты признаков размером $M \times N$ с количеством каналов P , базовым элементом для предсказания потенциального места расположения объекта является ядро с размером $M \times N \times P$. Это ядро соответствует набору k прямоугольников для соответствующей ячейки. Поэтому всего будет $(c + 4)k$ фильтров, применяемых к каждому возможному положению ограничивающих прямоугольников, т.е. на выходе каждого слоя будет вектор из $(c + 4)kMN$ значений.

Подавление не максимумов. На последнем этапе выполняется подавление не максимумов, т. е. из всего набора прямоугольников отбрасываются ограничивающие прямоугольники, имеющие слишком малую вероятность, а также не удовлетворяющие каким-либо другим (например — зависящим от специфики задачи) критериям. Комбинирование предсказаний для всех финально вычисленных областей для различных масштабов и соотношений сторон позволяет локализовать объекты разных размеров и форм.

В диссертации в качестве подавления не максимумов для набора ограничивающих прямоугольников используется следующий алгоритм. Сначала из общего набор отсеиваются все ограничивающие прямоугольники с вероятностью наличия в них того или иного объекта меньше 0,3 (в нашем случае рассматривается только один тип объектов — коленный сустав). Это позволяет отбросить большую часть всех некорректных или пустых областей.

Несмотря на предыдущий шаг, каждый объект на рентгенограмме может быть представлен несколькими ограничивающими прямоугольниками, которые с разной точностью будут обозначать искомые суставы (рис. 44). Для того, чтобы выбрать из них финальные области, необходимо выполнить дополнительную фильтрацию, которая состоит из следующего рекурсивного алгоритма:

1. из набора выбирается прямоугольник с наибольшей вероятностью наличия в нем объекта;
2. из оставшегося набора удаляются все прямоугольники, которые при сравнении с выбранным имеют коэффициент Жаккарда, больший чем 0,6;
3. из оставшихся выбирается следующий прямоугольник с наибольшей вероятностью и повторяется шаг 2, и так до тех пор, пока все прямоугольники не будут обработаны.

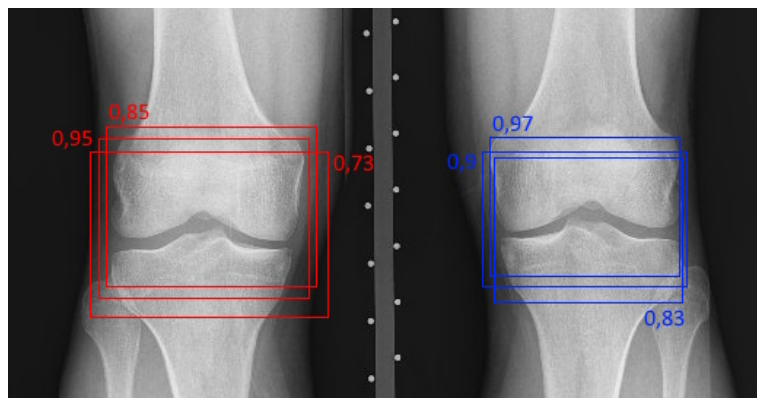


Рисунок 44 — Иллюстрация нескольких вариантов локализации для каждого сустава методом SSD (до подавления не максимумов)

Данный алгоритм позволяет получить по одному ограничивающему прямоугольнику на сустав, каждый из которых наиболее точно описывает соответствующий объект.

2.1.2 Архитектура MobileNet

После победы AlexNet в конкурсе ImageNet 2012-го года сверточные сети начали набирать все большую и большую популярность и находить применение в самых разнообразных задачах. Главным направлением улучшения качества работы сети было увеличение ее глубины и сложности. Все эти модификации, как

правило, делали сети более ресурсоемкими и требовательными к вычислительным мощностям оборудования, на котором они должны были применяться.

С развитием робототехники, беспилотных автомобилей и других, имеющих ограничения на имеющиеся вычислительные ресурсы, технологий, а также с приложением нейронных сетей к этим технологиям, появилась необходимость в поиске архитектур, которые смогут работать в условиях всех этих ограничений с приемлемой точностью и скоростью. Одна из таких архитектур с названием MobileNet была предложена в 2017 году в работе [30]. Основное назначение данной архитектуры — использование сверточных нейронных сетей в различных мобильных устройствах (смартфоны и т.п.), однако из-за своих характеристик она также находит применение во многих других областях.

Общая концепция. MobileNet построена из блоков, состоящих из так называемых разделимых по глубине сверток (depthwise separable convolutions), которые используются в сетях архитектуры Inception для уменьшения вычислительной сложности [26]. Традиционные свертки в нейронных сетях применяются одновременно к некоторой окрестности пикселя как в плоскости слоя, так и по всем каналам (если слой имеет глубину, большую единицы). Т.е. такой слой одновременно обрабатывает и пространственную, и межканальную информацию. Разделимые по глубине свертки опираются на предположение, что пространственную и межканальную информацию можно обрабатывать независимо друг от друга, сохраняя при этом обобщающую способность модели и уменьшая вычислительную сложность.

Для этого традиционные свертки заменяются на комбинацию фильтров размером $F \times F$, которые имеют глубину, равную 1, и применяются к каждому отдельному каналу слоя независимо друг от друга (depthwise) и фильтров размером 1×1 , которые обрабатывают межканальную корреляцию (pointwise). Такое разделение позволяет существенно уменьшить количество вычислений и итоговый размер обученной модели.

Рассмотрим обычный сверточный слой нейронной сети, который принимает на вход карту признаков размером $D_F \times D_F \times M$ и на выходе преобразует ее в карту признаков $D_F \times D_F \times N$, где D_F — это пространственный размер слоя (ширина и высота изображения), а M и N — количество каналов во входном и выходном изображении соответственно. В стандартном случае, для получения всех каналов выходного изображения требуется ко входному изображению применить N фильтров $D_K \times D_K \times M$, где D_K — это пространственный раз-

мер фильтра для свертки. Вычислительную сложность для такого слоя можно определить как

$$D_K \cdot D_K \cdot M \cdot N.$$

Если же разделить обычную свертку на два этапа — свертку каждого слоя отдельным одноканальным фильтром размера $D_K \times D_K$ и последующую свертку результата N фильтрами $1 \times 1 \times M$ — то общая вычислительная сложность слоя будет выглядеть следующим образом:

$$D_K \cdot D_K \cdot M + N \cdot M.$$

Используя полученные оценки, можем посчитать эффективность разделимых по глубине сверток по сравнению с традиционными свертками, применяемых в сверточных сетях:

$$\frac{D_K \cdot D_K \cdot M + N \cdot M}{D_K \cdot D_K \cdot M \cdot N} = \frac{1}{N} + \frac{1}{D_K^2}.$$

MobileNet использует фильтры размером 3×3 . Если взять в качестве примера сверточный слой с 32 входными и 64 выходными каналами, то количество параметров для такого слоя будет $32 \cdot 64 \cdot 3 \cdot 3 = 18432$. Если же вместо стандартной свертки по всем каналам воспользоваться описанной выше комбинацией сверток, но количество параметров будет равно $32 \cdot 64 + 64 \cdot 3 \cdot 3 = 2624$, т. е. параметров более чем в 7 раз меньше. При этом точность работы уменьшается незначительно, а в некоторых случаях даже превосходит более вычислительно емкие модели [30].

Описание архитектуры. Сеть с архитектурой MobileNet состоит из чередующихся разделяемых по глубине сверток с фильтрами размером 3×3 , которые были упомянуты в предыдущем разделе, и обычных сверток с фильтром 3×3 , за исключением первого слоя сети, который является обычным сверточным слоем с размером фильтра 3×3 (табл. 5). Каждый слой при этом сопровождается нормализацией батчей [24] и нелинейностью в виде ReLU, последний слой — полносвязный, за которым следует Softmax.

Таблица 5 — Архитектура сети MobileNet: Conv - обычные свертки, conv dw - свертки, разделимые по глубине (depthwise separable convolutions)

Слои	Вход	Размер фильтра, шаг
Conv	$224 \times 224 \times 3$	$3 \times 3 \times 3 \times 32, 2$
Conv dw	$112 \times 112 \times 32$	$3 \times 3 \times 32, 1$
Conv	$112 \times 112 \times 32$	$1 \times 1 \times 32 \times 64, 1$
Conv dw	$112 \times 112 \times 64$	$3 \times 3 \times 64, 2$
Conv	$56 \times 56 \times 64$	$1 \times 1 \times 64 \times 128, 1$
Conv dw	$56 \times 56 \times 128$	$3 \times 3 \times 128, 1$
Conv	$56 \times 56 \times 128$	$1 \times 1 \times 128 \times 128, 1$
Conv dw	$56 \times 56 \times 128$	$3 \times 3 \times 128, 2$
Conv	$28 \times 28 \times 128$	$1 \times 1 \times 128 \times 256, 1$
Conv dw	$28 \times 28 \times 256$	$3 \times 3 \times 256, 1$
Conv	$28 \times 28 \times 256$	$1 \times 1 \times 256 \times 256, 1$
Conv dw	$28 \times 28 \times 256$	$3 \times 3 \times 256, 2$
Conv	$14 \times 14 \times 256$	$1 \times 1 \times 256 \times 512, 1$
5 × Conv dw	$14 \times 14 \times 512$	$3 \times 3 \times 512, 1$
5 × Conv	$14 \times 14 \times 512$	$1 \times 1 \times 512 \times 512, 1$
Conv dw	$14 \times 14 \times 512$	$3 \times 3 \times 512, 2$
Conv	$7 \times 7 \times 512$	$1 \times 1 \times 512 \times 1024, 1$
Conv dw	$7 \times 7 \times 1024$	$3 \times 3 \times 1024, 2$
Conv	$7 \times 7 \times 1024$	$1 \times 1 \times 1024 \times 1024, 1$
Avg Pool	$7 \times 7 \times 1024$	Pool $7 \times 7, 1$
FC	$1 \times 1 \times 1024$	$1024 \times 1000, 1$
softmax	$1 \times 1 \times 1000$	Classifier

2.2 Эксперимент

Обучение. Обучение сети осуществлялась методом RMSProp (root mean square propagation [118]) с коэффициентом L2-регуляризации (weight decay), равным 0,9, моментом 0,9 и значением параметра $\epsilon = 1$ (число, добавляемое в знаменатель выражения для стабильности). Размер батча, исходя из доступных вычислительных ресурсов, был выбран равным 24. Стартовый коэффициент

скорости обучения (learning rate) — $4e - 3$, который уменьшался на 5% каждые 40000 шагов (примерно 200 эпох).

Исходный набор данных был разбит на обучающую и тестовую выборки в соотношении 70% и 30% соответственно. Таким образом, финальный размер обучающей и тестовой подвыборок составлял 3357 и 1439 изображений. Чтобы расширить доступный набор изображений обучения, была использована аугментация данных. Аугментация, в виде случайного зеркального отражения по горизонтали и вырезания случайной части изображения с последующим масштабированием до исходного размера (300×300), выполнялась «на лету», в процессе обучения.

Оценка результатов. В таблице 6 приведено сравнение точности работы предложенного метода и методов из ранее опубликованных работ. Числа в таблице обозначают процент корректно локализованных коленных суставов для соответствующего коэффициента Жаккарда. Так, приведены данные для коэффициентов 0,25, 0,5 и 0,75, а также среднее значение этого коэффициента (Mean). Кроме того, для каждого исследования приводится количество изображений тестового датасета (N), на котором были получены соответствующие результаты — при этом на каждом изображении как правило имеется два коленных сустава, если не указано иное. Прочерк означает, что соответствующие данные недоступны.

Таблица 6 — Сравнение методов автоматического детектирования, основанное на коэффициенте Жаккарда (J)

Метод	$J \geq 0,25$	$J \geq 0,5$	$J \geq 0,75$	Mean	N
Гист. + центр масс [47]	-	-	-	0,849	98
Гист. + Gabor [47]	-	-	-	0,90	98
Region Proposal [53]	-	-	-	0,84	473
Template Matching [60; 82]	~54,4%	8,3%	-	0,1	4496
Linear SVM + Sobel [60]	~81,8%	38,6%	-	0,36	4496
FCN, OAI [65]	100%	99,9%	89,2%	0,83	1300
FCN, MOST [65]	99,5%	98,4%	85,0%	0,81	900
FCN, OAI+MOST [65]	99,9%	99,9%	91,4%	0,83	2200
YOLO [67]	-	-	-	0,82	200
YOLOv2 [68]	-	99,9%	92,2%	0,858	828
SSD, наш метод	100%	100%	94,1%	0,844	1439

Методы, использующие гистограмму распределения яркости по строкам или столбцам [47], требуют предварительного этапа разделения левого и правого сустава рентгенограммы, т.е. могут работать лишь с изображениями, на которых имеется ровно один сустав в правильной ориентации. Это накладывает определенные ограничения на область их применения.

Для работы [67] в таблице приведена точность на *тренировочном* датасете из 200 изображений, т.к. методика оценки выбранного метода локализации отличается от общепринятой — после обучения на 200 изображениях, было проведено тестирование на 4315 изображениях, при этом оценивалось не точность локализации, а количество суставов на изображении, положение локализованных суставов, их ширина и т.п. Данная методика не позволяет корректно оценить качество работы метода в сравнении с остальными.

Метод сопоставления с шаблоном для локализации коленного сустава (Template Matching) изначально был приведен в работе [82], однако в данной работе отсутствовали результаты тестирования предложенного метода локализации. В работе [60] данный метод был использован для сравнения с алгоритмом скользящего окна в сочетании с классификатором SVM, и финальные результаты тестирования для определения индекса Жаккарда приведены именно там. Особенностью подобных методов является то, что они не требуют большого количества изображений для обучения, или же обучающая выборка состоит лишь из участков рентгенограмм, поэтому финальное тестирование проводилось на полном размеченном датасете из 4496 изображений.

Сравнение результатов показывает, что методы, основанные на использовании сверточных нейронных сетей, являются более универсальными и показывают более высокую точность детектирования, нежели другие подходы. При этом в плане быстродействия, в современных реалиях, более простые, но менее точные методы локализации не имеют никаких серьезных преимуществ перед нейронными сетями, а в случае пакетной обработки на графическом ускорителе использование нейронных сетей представляется более предпочтительным вариантом.

Также по результатам экспериментов можно сделать вывод, что использование для вычисления набора признаков предтренированной сети в задаче локализации позволяет достичь большей точности по сравнению с сетью, обученной с нуля (качество локализации для YOLO, YOLOv2 и SSD значительно

превосходит качество локализации обученных с нуля полностью сверточных нейронных сетей FCN).

2.3 Выводы

В соответствии с поставленными задачами, во второй главе были получены следующие теоретические и практические результаты.

1. Предложено решение поставленной задачи локализации коленного сустава при помощи нейросетевой архитектуры Single Shot Detector (SSD), состоящей из двух частей: извлечение карты признаков из входного изображения и поиск результирующих областей на изображении по полученной карте признаков. В качестве способа вычисления карты признаков предложено использовать легковесную и быструю нейронную сеть архитектуры MobileNet, предобученную на датасете COCO.
2. Осуществлена программная реализация и проведен процесс обучения предложенной архитектуры на размеченном вручную наборе данных, в основе которого лежат данные датасета OAI.
3. Проведены вычислительные эксперименты и определены количественные метрики качества локализации области коленного сустава на рентгенограмме предложенным методом, выполнено сравнение с существующими методами.

Результаты данной главы опубликованы в работе [147].

Глава 3. Разработка и исследование метода автоматической классификации остеоартрита

В данной главе описано решение задачи автоматической классификации остеоартрита при помощи сверточных нейронных сетей различных архитектур. Каждый классификатор работает непосредственно с изображениями коленного сустава размером 224×224 или 299×299 (в зависимости от рассматриваемой архитектуры), которые в свою очередь извлекаются из исходных рентгенограмм при помощи сети SSD, описанной в главе 2.

В конце главы приведены численные оценки качества работы классификаторов, основанных на использовании различных архитектур сверточных нейронных сетей, полученные в результате экспериментов над набором данных OAI. Эксперимент проводился для различных функций потерь, используемых при обучении, и разного количества выходных классов остеоартрита (полная пятиклассовая шкала Келлгрена-Лоуренса, сокращенная шкала и бинарная классификация, определяющая наличие признаков остеоартрита любой стадии на рентгенограмме).

При этом, для каждой модели и каждой функции потерь проведено три эксперимента, различающиеся между собой начальным значением генератора случайных чисел, что позволяет обучать одни и те же модели на разных конфигурациях данных из обучающей выборки, т.е. выполнить своего рода кросс-валидацию. В итоговых таблицах представлены усредненные данные тестирования обученных моделей, а в качестве финальных версий выбраны те наборы весов, при которых была получена наилучшая оценка точности на валидационной выборке.

3.1 Схема и параметры обучения нейронных сетей

Во всех экспериментах (если не оговорено иное) обучение моделей осуществлялось методом Adam (the adaptive moment estimation optimizer [119]) со скоростью обучения, равной $1e - 3$, коэффициентом L2-регуляризации $1e - 4$ и размером батча 32. В процессе обучения скорость обучения уменьшалась на

5% каждые 5 эпох. Всего для обучения каждой модели оказалось достаточно 75 эпох — выше этих значений, из-за недостаточного количества данных (несмотря на применяемую аугментацию), сети начинали переобучаться на тренировочном множестве.

3.1.1 Функция потерь

Перекрестная энтропия. В качестве основной функции потерь использовалась категориальная перекрестная энтропия (categorical cross-entropy loss). В каноническом виде перекрестная энтропия описывается формулой

$$CE = - \sum_i^C t_i \log s_i,$$

где t_i и s_i — истинные и предсказанные нейронной сетью метки для каждого класса i из множества всех классов C . Категориальная перекрестная энтропия отличается от канонической тем, что в этом случае сначала применяется softmax, и лишь затем функция перекрестной энтропии, т.е.

$$f(s)_i = \frac{e^{s_i}}{\sum_j^C e^{s_j}}, CE = - \sum_i^C t_i \log f(s)_i. \quad (1)$$

где $f(s)_i$ — функция активации softmax для i -го класса. Во многих (на самом деле — в большинстве) случаях мультиклассовой классификации результат работы классификатора для некоторого класса C_p описывается вектором t , в котором только одно ненулевое значение t_p для целевого класса, а метки остальных классов в выходном векторе при этом равны нулю. Таким образом, исключив нулевые элементы выходного вектора в равенстве 1, мы можем переписать его в виде

$$CE = - \log \frac{e^{s_p}}{\sum_j^C e^{s_j}},$$

где s_p является результатом работы сверточной сети для *положительного* (т.е. истинного) класса соответствующего вектора t .

Регулируемая порядковая функция потерь. Особенностью предсказания степени остеоартрита при помощи автоматического классификатора является то, что ошибки в предсказании между классами не эквивалентны — к примеру, ошибка между классами 0 и 1 менее значима, чем между классами 0 и 4. При использовании перекрестной энтропии данное обстоятельство не учитывается и все ошибки считаются равноценными.

Чтобы учитывать порядок классов, выходное распределение вероятностей классов (после слоя softmax) для некоторого изображения с истинным классом m должно удовлетворять двум свойствам:

1. p_m должна быть как можно ближе к единице;
2. для $k \in \{0, \dots, n - 1\} \setminus \{m\}$, чем $|k - m|$ больше, тем p_k должно быть меньше (здесь n — общее количество классов).

В работе [68] предлагается новая функция потерь, которая позволяет учитывать порядок классов и по разному штрафовать ошибки между разными классами. Для этого вводится регулируемая порядковая матрица W (adjustable ordinal matrix) размером $n \times n$, которая содержит значения штрафов между предсказанным и истинным классами. Так, $w_{i,j}$ содержит значение штрафа между предсказанным классом j относительно класса i . Пример порядковой матрицы для обучения классификации остеоартрита по пятиклассовой шкале Келлгрена-Лоуренса приведен на рис. 45.

	кл-0	кл-1	кл-2	кл-3	кл-4
кл-1	1	3	5	7	9
кл-1	3	1	3	5	7
кл-2	5	3	1	3	5
кл-3	7	5	3	1	3
кл-4	9	7	5	3	1

Рисунок 45 — Пример порядковой матрицы W из [68], которая также используется и в данной работе

Значение штрафа при соответствии классов (главная диагональ W) равны единице, штраф при несоответствии классов больше единицы. Обозначим вектор штрафов для класса m через $w_{:,m}$, и тогда функцию потерь можно представить в виде

$$ORD = \sum_{i=0}^{n-1} w_{i,m} \cdot q_i, \quad (2)$$

где m — истинная степень ОА для входного изображения, $q_i = p_i$, если $i \neq m$ и $q_i = 1 - p_i$ в противном случае. Минимизация функции из равенства 2 требует, чтобы p_m была как можно ближе к 1, и при этом чем больше «расстояние» между классами — тем более высокое значение будет принимать функция потерь (из-за штрафа в матрице W).

Для упрощения реализации авторы предлагают переформулировать матрицу штрафов W таким образом, чтобы в новой матрице $\bar{w}_{i,j} = 0$ при $i = j$ и $\bar{w}_{i,j} = w_{i,j} + 1$ в противном случае. И тогда функция будет иметь вид

$$ORD = \sum_{i=0}^{n-1} \bar{w}_{i,m} \cdot p_i. \quad (3)$$

В том, что равенства 2 и 3 эквивалентны, можно убедиться, просто подставив конкретные значения m и n в формулу 3. Таким образом, мы имеем более удобное для реализации представление функции потерь 3, которое можно использовать непосредственно с выходом слоя softmax нейронной сети (вектором p).

3.1.2 Аугментация

Для устранения переобучения используются различные подходы. На уровне модели применяется уменьшение количества обучаемых параметров или регуляризация весов в процессе обучения [120]. На уровне данных одним из основных методов является аугментация, которая заключается в применении к исходным данным разного рода преобразований, не меняющих основную информацию, содержащуюся на изображении, но слегка ее искажающих, что позволяет получить новые примеры на основе имеющихся.

В данной работе применялась аугментация «на лету», которая заключается в применении трансформирующих преобразований непосредственно перед подачей данных на вход нейронной сети. Преимуществом данного подхода является то, что не увеличивается размер основной обучающей выборки и тем

самым увеличивается пропускная способность памяти, которая во многих случаях является узким местом.

В качестве преобразований в данной работе использовалось случайное зеркальное отражение по горизонтали, случайное вращение изображений ($-10..10^\circ$), масштабирование ($0,9..1,1$), изменение яркости и насыщенности цветовой гаммы. Все преобразования применялись в случайном порядке с вероятностью 50%. После выполнения случайного набора преобразований изображения масштабировались до рабочего размера 224×224 или 299×299 , в зависимости от исследуемой архитектуры.

В процессе обучения в качестве финальной модели выбиралась модель, достигшая наибольшей точности на валидационной выборке среди всех эпох.

3.2 Разработка метода автоматической классификации остеоартрита коленного сустава

В начале исследований было решено проверить эффективность универсального метода классификации под названием WND-CHARM, который ранее уже использовался для обработки медицинских изображений [80]. Описание данного метода и его особенности кратко представлены в первой главе в разделе с описанием существующих методов.

Далее была выполнена реализация простой сверточной нейронной сети, которая была предложена в работе [65] и является одной из первых сверточных нейронных сетей, использовавшихся для решения задачи автоматической классификации остеоартрита коленного сустава по рентгенограмме.

После анализа полученных результатов, а также изучения литературы, было принято решение провести исследование эффективности основных известных архитектур и архитектурных решений с целью найти те модели, которые показывают наилучшие результаты при решении поставленной задачи. Для всех архитектур имеется описание структуры использовавшихся моделей, описание процесса и параметров обучения, а также результаты тестирования на тестовой выборке (см. Приложение Б).

3.2.1 Классификация остеоартрита методом WND-CHARM

Метод WND-CHARM основан на выделении из изображений некоторых векторов-признаков, на основе которых при помощи модифицированного метода ближайших соседей определяется принадлежность к тому или иному классу. В качестве эксперимента метод был опробован на имеющемся наборе данных из нескольких тысяч изображений различных стадий остеоартрита. При этом в качестве изображений для анализа использовались непосредственно изображения области коленного сустава. В таблице 7 показано распределение количества изображений из разных классов, участвовавших в обучении классификатора и его тестировании.

Таблица 7 — Распределение изображений, использовавшихся для эксперимента с WND-CHARM, по стадиям остеоартрита шкалы Келлгрена-Лоуренса

	KL-0	KL-1	KL-2	KL-3	KL-4	Всего
Тренировка	2602	1206	1779	926	219	6732
Тестирование	867	402	593	308	73	2243

Для стабильности было проведено несколько экспериментов с различными разбиениями исходного датасета на тренировочную и тестовую выборки. Кроме того, эксперимент проводился над несколькими вариантами размера изображений — использовался как оригинальный размер, так и масштабирование образцов до 75% и 50% от базового разрешения.

В результате работы алгоритма, из каждого изображения было извлечено 4024 признака (features). В эксперименте с датасетом, содержащим уменьшенные вдвое (0,5) от исходного размера изображения, для основной классификации были отобраны 403 характеристики, имеющие наибольшую оценку Фишера. Для коэффициентов масштабирования 0,75 и 1, также основываясь на оценке Фишера, основной набор признаков был уменьшен с 4024 до 805 штук. Результаты экспериментов в виде среднеклассовой точности для различных степеней уменьшения размерности исходного изображения представлены в таблице 8.

Во всех экспериментах наибольшее влияние на финальный результат классификации оказали группы признаков, относящиеся к коэффициентам полиномов Цернике (Zernike), аппроксимирующие не только исходное изображение, но и некоторые его преобразования — карту границ (результат применения к

Таблица 8 — Точность классификации методом WND-CHARM для различных вариантов понижения оригинальной размерности (1.0 — оригинальный размер изображений)

Понижение размерности	Количество признаков	Точность
0,5	403 / 4024	38%
0,75	805 / 4024	38%
1,0	805 / 4024	37%

исходному изображению оператора Прюитт), вейвлет-преобразование и вейвлет-преобразование карты границ. Кроме того, большое значение также оказали признаки, полученные из преобразований Чебышева над картой границ и первые четыре момента из исходного изображения, карты границ и вейвлет-преобразования карты границ.

На рис. 46 проиллюстрированы матрицы ошибки для всех запусков классификатора. При этом, в соответствующих ячейках матрицы просуммированы результаты по всем разбиениям.

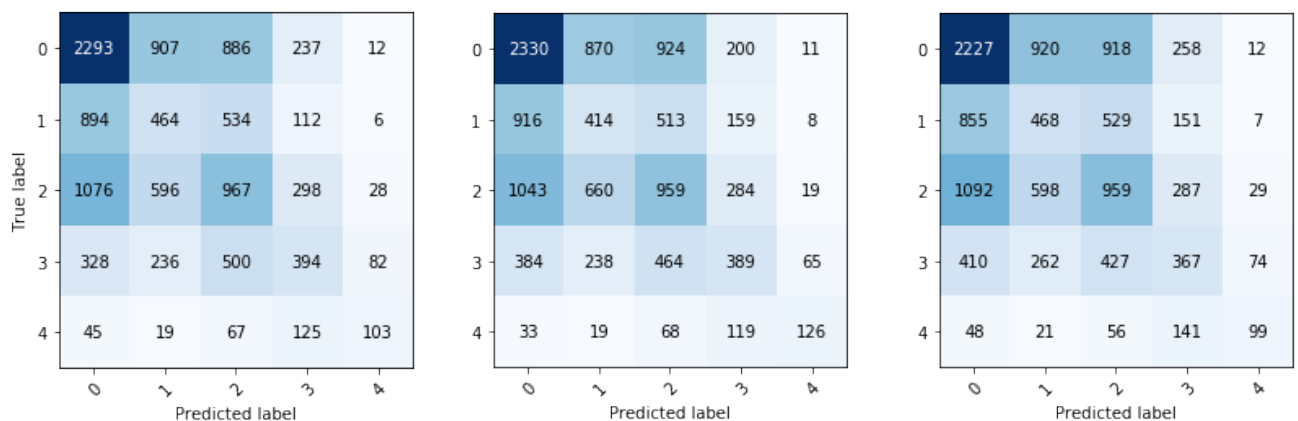


Рисунок 46 — Визуализация матриц ошибок для классификаторов WND-CHARM с коэффициентом масштабирования 0,5, 0,75 и 1,0 от исходного размера изображения соответственно (слева направо)

В таблице 9 представлены подробные метрики качества классификации методом WND-CHARM для каждого класса шкалы KL, соответствующие показанным выше матрицам ошибок. Приведены результаты для всех протестированных вариантов понижения размерности.

Из таблицы видно, что наивысшая точность классификации (из-за особенностей вычисления совпадает с параметром *recall*) методом WND-CHARM

Таблица 9 — Метрики качества работы метода WND-CHARM для всех использовавшихся в эксперименте коэффициентов понижения размерности

ОА	0.5			0.75			1.0		
	Prec.	Recall	F_1	Prec.	Recall	F_1	Prec.	Recall	F_1
0	0,49	0,53	0,51	0,50	0,54	0,52	0,48	0,51	0,50
1	0,21	0,23	0,22	0,19	0,21	0,20	0,21	0,23	0,22
2	0,33	0,33	0,33	0,33	0,32	0,33	0,33	0,32	0,33
3	0,34	0,26	0,29	0,34	0,25	0,29	0,31	0,24	0,27
4	0,45	0,29	0,35	0,55	0,35	0,42	0,45	0,27	0,34
Ср.	0,36	0,33	0,34	0,38	0,33	0,35	0,36	0,31	0,33

получена для класса 0 шкалы Келлгрена-Лоуренса и составляет чуть более 50%. Для остальных классов точность классификации заметно ниже.

Результаты применения WND-CHARM к решению поставленной задачи в данном исследовании сопоставимы с результатами применения этого же метода в других работах [65; 82]. Также, по полученной точности работы метода, можно сделать вывод, что используемого в методе набора признаков недостаточно для получения приемлемых результатов классификации. Низкая точность классификации может быть обусловлена тем, что выбранные признаки учитывают лишь преимущественно текстурные особенности изображения и не различают определенные физические паттерны, сопровождающие ОА, такие как наличие остеофитов, деформация костей и сужение межсуставного пространства.

3.2.2 Классификация при помощи простой сверточной нейронная сеть

Одной из первых работ, в которых было предложено полноценное использование сверточных сетей для решения задачи классификации ОА, можно считать нейронную сеть из работы [65]. Авторы данной работы во многом вдохновлялись победителем конкурса ImageNet 2012 года — архитектурой AlexNet [16]. Архитектура предлагаемой ими сети представлена в таблице 10.

После каждого сверточного или полносвязного слоев обязательно использовалась нормализация батчей, которая отсутствовала в оригинальной сети AlexNet, и нелинейность в виде ReLU. Кроме того, для каждого сверточного слоя

Таблица 10 — Архитектура сети из работы [65]. Размер входного изображения — 300×200 , аббревиатура conv означает сверточный слой, pool — слой пуллинга, fc — полносвязный слой

Слои	Выход	Параметры слоя
conv1	150x100	11x11x32 conv, stride 2
pool1	75x50	3x3 max pool, stride 2
conv2	75x50	5x5x64 conv, stride 1
pool2	32x25	3x3 max pool, stride 2
conv3	32x25	3x3x96 conv, stride 1
pool3	16x13	3x3 max pool, stride 2
conv4	16x13	3x3x128 conv, stride 1
pool4	8x7	3x3 max pool, stride 2
fc1	1024	7168x1024
fc2	5	1024x5

применялось расширение размера с каждой стороны изображения (padding), равное половине размера фильтра.

Комбинирование нескольких сетей для получения финального результата обычно позволяет повысить точность и уменьшить финальную ошибку, однако обучение большого количества сетей может быть чрезвычайно ресурсоемкой задачей. Для решения этой проблемы была придумана техника, называемая Dropout [121]. При использовании этой техники, при тренировке слоя сети, для которой эта техника применяется, через каждый нейрон сети сигнал проходит с определенной вероятностью, и при этом отброшенные (dropped out) нейроны не участвуют в процессе обратного распространения ошибок. Т.е. при каждой итерации происходит своего рода модификация сети и изменение связей внутри нее. Это эквивалентно обучению экспоненциально огромного числа нейронных сетей похожих, но слегка отличающихся, архитектур, которые как бы делятся между собой весами.

Чтобы избежать переобучения, к последнему сверточному слою (conv4) применяется техника Dropout с параметром 0,2, а также Dropout с параметром 0,5 к последнему полносвязному слою (fc2). К выходу последнего слоя (размером 5, по количеству классов), для получения вероятностей каждого из классов, применяется Softmax.

Обучение. Был проведен эксперимент по обучению описанной выше архитектуры с тем изменением, что сеть применялась не к изображениям размера 300×200 , а к изображениям размера 224×224 — для удобства дальнейшего сравнения точности различных архитектур. Также, если в оригинальной работе L2-регуляризация применялась только к последним слоям модели, в эксперименте такая регуляризация была применена ко всем слоям, что в теории должно повысить точность данной сети.

Для обучения сети, как и в оригинальной работе, использовалась минимизация категориальной перекрестной энтропии в качестве функции потерь при помощи метода Нестерова (стохастический градиентный спуск с импульсом). В качестве данных для обучающей и тестовой выборок использовался датасет, описанный в одном из предыдущих разделов 3.1. Обучалась сеть с нуля, т.е. никаких заранее обученных весов не использовалось, а начальные веса были сгенерированы случайным образом.

Эксперименты и результаты. В оригинальной работе авторам удалось добиться точности в **60,3%** при тренировке сети на классификацию. После модификации исходной сети — а именно добавления в функцию потерь учет среднеквадратичного отклонения для учета регрессии помимо классификации классов — точность сети возросла до **63,6%**.

В эксперименте, в котором сеть обучалась только для классификации содержимого входных изображений на 5 классов шкалы Келлгрена-Лоуренса, была получена точность в **61,78%**. В таблице 11 приведены подробные метрики качества работы обученной сети на тестовом множестве в виде таких параметров, как точность, полнота и F_1 -мера, а также аналогичные параметры из оригинальной работы [65]. Т. к. в нашем случае классы в датасете не сбалансированы, это наиболее подходящий способ оценки результатов.

На рис. 47 представлена матрица ошибок из работы [65] и ROC-кривая для каждой из степеней шкалы Келлгрена-Лоуренса, соответствующая этой матрице ошибок. Результаты показывают, что из-за слабых отличительных черт изображений классов 0 и 1, алгоритм очень плохо распознает рентгенограммы со степенью ОА, равной 1 по шкале KL. Данный факт отмечался и в других работах.

На рис. 48 изображен график обучения сети на тренировочной (оранжевый цвет) и валидационной (синий цвет) выборках, где по оси X отложен номер эпохи обучения, а по оси Y — ассурасу. График приведен для обучения методом

Таблица 11 — Метрики результатов тренировки сети как для модели из проведенного эксперимента, так и для аналогичной модели из оригинальной работы

ОА	Эксперимент			Работа [65]		
	Precision	Recall	F1	Precision	Recall	F1
0	0,58	0,94	0,72	0,63	0,82	0,71
1	0,33	0,00	0,01	0,25	0,04	0,06
2	0,64	0,51	0,56	0,47	0,57	0,51
3	0,75	0,77	0,76	0,76	0,71	0,73
4	0,78	0,49	0,60	0,78	0,77	0,77
Среднее	0,58	0,62	0,55	0,56	0,60	0,56

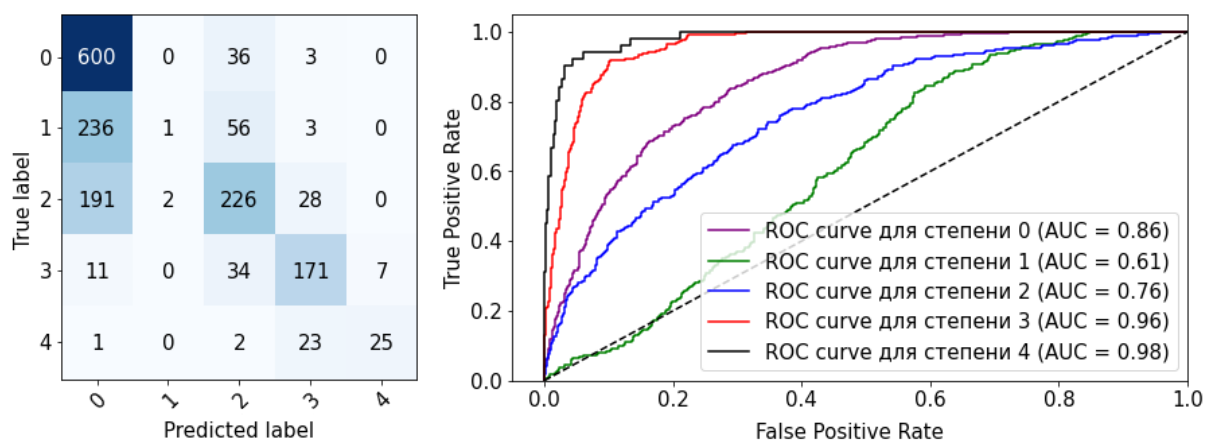


Рисунок 47 — Матрица ошибок (слева) и ROC-кривая (справа), полученные в результате экспериментов для модели из работы [65]. На графике ROC-кривой также указаны значения площади под кривой (AUC) для каждой из степеней ОА

Нестерова. Стоит отметить, что в случае использования оптимизатора Adam, скорость обучения сети повышалась более чем в два раза (сопоставимый по точности результат был получен уже в районе 100-й эпохи), при этом точность работы сети также незначительно повышалась.

Результаты оценки классификатора, основанного на использовании нейронной сети, показывают, что подобный подход позволяет достичь гораздо более высокой точности, нежели использование методов, не основанных на нейросетевых технологиях (что показано выше и прослеживается в первой главе в секции с обзором существующих методов автоматической классификации). Далее будут представлены результаты экспериментов по оценке применимости различных архитектур сверточных нейронных сетей к решению данной задачи.

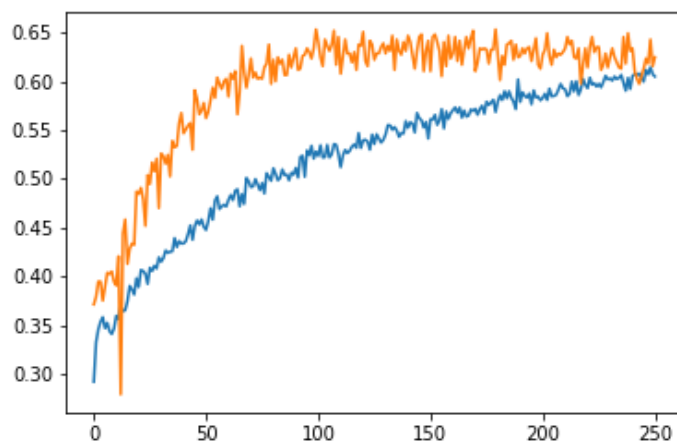


Рисунок 48 — График обучения сети из работы [65]. По оси X отложен номер эпохи, по оси Y — точность. Оранжевый график — точность работы модели на тренировочной выборке, синий — на валидационной

3.2.3 Использование классических архитектур сверточных нейронных сетей и новых архитектурных решений

Все предложенные ранее традиционные архитектуры сверточных нейронных сетей (ResNet, Inception и т.д.) первоначально были разработаны для конкурса ImageNet Large-Scale Visual Recognition Challenge, в котором им предлагалось классифицировать изображения 1000 различных классов, таких как «кот», «собака», «самолет» и т.п. Из-за этого для каждой из архитектур имеются обученные на датасете ImageNet модели, которые могут решать только конкретную узкоспециализированную задачу распознавания 1000 классов.

Обучение сверточных нейронных сетей с нуля требует использования большого количества обучающих данных, и финальная точность работы таких моделей напрямую зависит от размера обучающей выборки. Особенно это касается случая сложных архитектур. Проведенные в рамках работы эксперименты показывают, что имеющихся нескольких тысяч изображений рентгенограмм коленного сустава недостаточно для обучения с нуля таких глубоких архитектур как ResNet, DenseNet и т.п. Кроме того, если провести аналогию с обучением человека, можно отметить, что в противоположность обучению решения каждой задачи с нуля человек опирается на существующий опыт, который не обязательно должен быть связан с этой конкретной задачей.

В случае с задачей классификации остеоартрита в используемой шкале Келлгрена-Лоуренса имеется (в базовом варианте) всего 5 классов. Для того,

чтобы использовать уже обученные веса существующих моделей на новой задаче, используют технику *переноса обучения* (или Transfer Learning, [122]), или дообучения нейронной сети, которая ранее была обучена на одном датасете, на другом наборе данных. Таким образом, происходит своего рода перенос знаний из исходной задачи к целевой.

Чтобы адаптировать готовую сеть, предназначенную для классификации изображений определенного набора классов, для работы с новым набором классов (в данном случае 5, 4 и 2 — в зависимости от используемой шкалы оценки ОА), последний, отвечающий непосредственно за классификацию полносвязный слой этой сети заменяется на новый полносвязный слой с таким же количеством входов, но с 5 выходами. Далее возможно несколько различных вариантов действий.

Первый вариант может быть использован, если исходная и целевая задачи очень похожи и имеют схожие по смыслу признаки различных классов (к примеру, к существующим 1000 классов нужно добавить несколько новых классов), и заключается в заморозке весов сверточных слоев и обучении исключительно весов последнего полносвязного слоя. Это позволяет существенно ускорить процесс дообучения модели, т.к. в процессе тренировки меняются лишь очень малое (по сравнению с общим количеством версов в модели) количество параметров сети. Эксперименты показали, что использование подобного подхода для адаптации сверточных сетей, обученных на датасете ImageNet, к решению задачи классификации остеоартрита, не позволяет добиться какого-то существенного улучшения качества сети по сравнению с обучением с нуля.

Альтернативным вариантом является дообучение всех весов модели, а не только весов нового слоя. Этот подход целесообразен, если набор классов (и признаков этих классов) исходной и целевой задачи сильно различаются. В дальнейшем, именно данная техника переноса обучения будет использована для большинства упомянутых архитектур (если не оговорено иное).

Полная шкала Келлгрена-Лоуренса

Процесс обучения для всех моделей происходил на одинаковых данных и в одних и тех же начальных условиях (одно и то же базовое значение гене-

ратора случайных чисел и др.). Результаты тестирования обученных сетей для полной шкалы Келлгрена-Лоуренса, состоящей из пяти классов, представлены в таблице 12.

Таблица 12 — Среднеклассовые метрики качества работы различных классификаторов для разных функций потерь (усредненные по трем экспериментам)

Архитектура	Cross-Entropy Loss,%			Ordinal Loss,%		
	Precision	Recall	F1	Precision	Recall	F1
Antony [65]	60,47	65,36	60,25	55,37	58,57	54,33
ResNet-18	61,94	65,24	62,54	65,75	66,69	65,96
ResNet-34	65,02	66,81	64,78	64,86	65,56	64,94
ResNet-50	64,04	66,43	64,47	64,72	66,20	65,14
ResNeXt-50	64,96	67,33	65,45	66,16	66,34	65,96
Inception3	67,56	69,93	68,02	68,61	69,08	68,27
Inception-ResnetV2	67,06	70,43	67,67	65,69	67,39	65,32
Xception	67,24	69,24	66,78	68,29	69,18	68,29
DenseNet-121	64,38	66,73	65,07	66,43	66,69	66,21
SE-Inception3	66,85	69,59	67,33	67,92	68,18	67,77
SE-ResNet-18	64,18	66,02	63,84	63,67	64,19	63,64
SE-ResNet-34	64,98	67,11	65,15	65,35	66,67	65,63
SE-ResNet-50	65,89	68,96	65,95	66,15	66,65	66,11
SE-ResNeXt-50	68,39	71,18	68,80	69,56	69,83	69,27
SE-Xception	67,81	69,14	67,72	67,76	68,84	67,45
SE-DenseNet-121	68,16	69,99	68,40	67,97	69,22	67,88

Как было упомянуто в разделе 1.4, параметры *recall* и *accuracy* для используемого метода оценки равны. Таким образом, наилучшая точность классификации в экспериментах с полной шкалой Келлгрена-Лоуренса была получена для архитектур **SE-ResNeXt-50** и **Inception-ResnetV2**, и составила 71,18% и 70,43% соответственно. Также достаточно высокий (по сравнению с другими архитектурами) результат был получен в экспериментах по обучению сетей SE-DenseNet-121 и Inception3 с точностью на тестовой выборке, равной 69,99% и 69,93%.

При использовании в качестве функции потерь Ordinal Loss, вопреки заявленному в исходной работе [68], финальная точность для всех упомянутых

выше архитектур оказалась меньше точности, полученной обучением при помощи стандартной категориальной перекрестной энтропии.

Эффект от использования SE-блоков для всех базовых архитектур показан на изображении 49. Можно отметить, что значительное улучшение качества работы (как для моделей, обучение которых выполнялось при помощи перекрестной энтропии, так и для моделей, обученных при использовании порядковой функции потерь) наблюдается только для архитектур SE-ResNeXt-50 и SE-DenseNet-121, в остальных же случаях улучшение либо незначительно, либо полностью отсутствует.

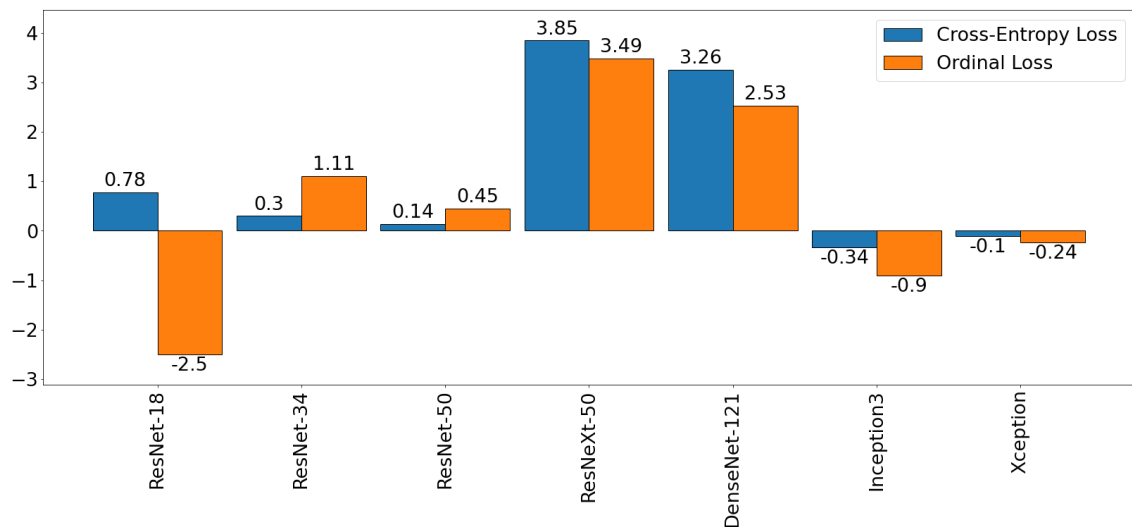


Рисунок 49 — Влияние блоков сжатия и возбуждения на качество работы сверточных сетей традиционных архитектур для полной шкалы Келлгрена-Лоуренса

На рисунке 50 представлена визуализация матриц ошибок для архитектур SE-ResNeXt-50 и Inception-ResnetV2, а в таблице 13 — рассчитанные по ней подробные метрики для каждого из классов остеоартрита. Можно заметить, что значения всех метрик (precision, recall, f-score) на пограничном классе 1 более чем в два раза меньше точности на оставшихся классах. Особенно данная негативная динамика наблюдается для метрики recall, которая, как упоминалось выше, в условиях данных методов оценки эквивалентна метрике ассигасу. Данная ситуация аналогичным образом повторяется на всех остальных архитектурах, а также в более ранних исследованиях ([53],[60],[65], [97], [68] и др.).

Во многом это обусловлено неоднозначностью шкалы Келлгрена-Лоуренса и слишком размытыми критериями для различия между нулевым-первым, и первым-вторым классами предложенной шкалы. Учитывая, что нулевой и первый классы ОА в медицинской практике считаются эквивалентными и трактуются

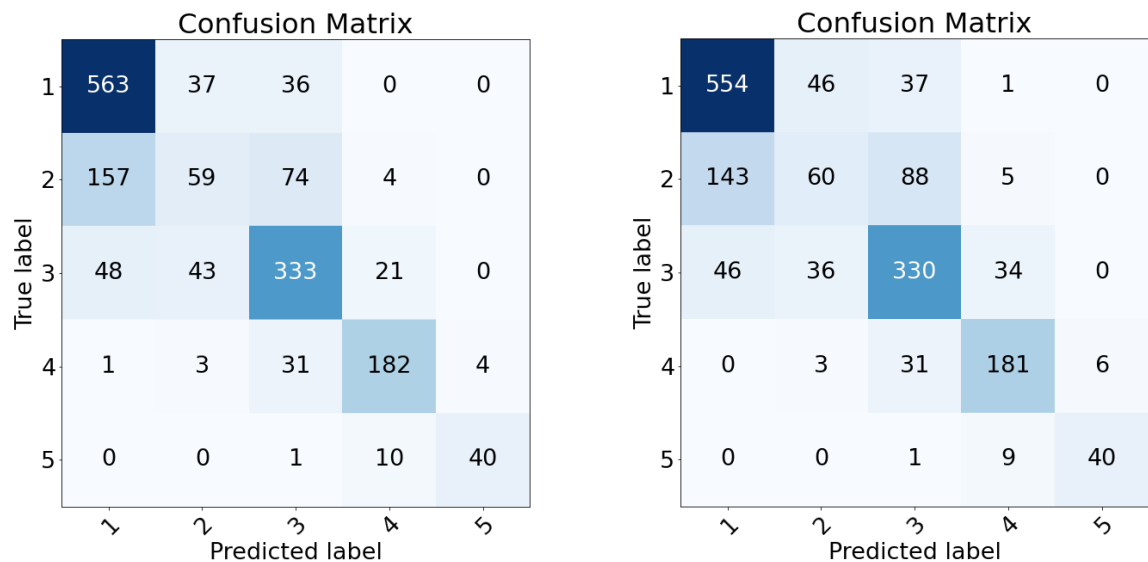


Рисунок 50 — Визуализация матриц ошибок для SE-ResNeXt-50 (слева) и Inception-ResnetV2 (справа)

Таблица 13 — Подробные метрики качества работы классификаторов, показавших наилучшие результаты работы для каждого из классов (усредненные по трем экспериментам)

OA	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
SE-ResNeXt-50						
0	0,73	0,88	0,80	0,76	0,85	0,80
1	0,41	0,20	0,27	0,39	0,34	0,36
2	0,70	0,75	0,72	0,71	0,67	0,69
3	0,84	0,82	0,83	0,85	0,76	0,80
4	0,90	0,78	0,84	0,83	0,80	0,81
Inception-ResnetV2						
0	0,75	0,87	0,80	0,71	0,90	0,79
1	0,36	0,20	0,24	0,34	0,24	0,27
2	0,68	0,74	0,71	0,71	0,57	0,63
3	0,80	0,81	0,80	0,76	0,81	0,78
4	0,87	0,80	0,82	0,90	0,72	0,79

как отсутствие остеоартрита, после анализа результатов работы всех классификаторов был проведен эксперимент с обучением этих классификаторов на диагностику ОА по сокращенной шкале Келлгрена-Лоуренса, в которой нулевой и первый классы объединены в один, означающий отсутствие патологий.

Сокращенная шкала Келлгрена-Лоуренса

Повторная серия экспериментов была проведена в тех же самых условиях и при тех же самых значениях гиперпараметров, что и в первом случае. Итоговые среднеклассовые метрики качества для классификаторов ОА, работающих со шкалой из четырех классов, представлены в таблице 14.

Таблица 14 — Среднеклассовые метрики качества работы различных классификаторов для разных функций потерь (усредненные по трем экспериментам)

Архитектура	Cross-Entropy Loss, %			Ordinal Loss, %		
	Precision	Recall	F1	Precision	Recall	F1
Antony [65]	77,26	77,90	76,93	69,69	72,25	69,46
ResNet-18	79,06	79,63	78,68	78,73	79,47	78,52
ResNet-34	78,76	79,39	78,57	80,08	80,27	79,69
ResNet-50	79,78	80,35	79,56	79,83	80,39	79,82
ResNeXt	79,98	80,48	79,55	77,59	78,78	77,69
Inception3	82,09	82,47	81,90	80,14	80,09	79,56
Inception-ResNet	80,26	80,76	80,04	80,98	81,32	80,62
Xception	81,33	81,50	81,22	80,64	80,94	80,58
DenseNet-121	80,28	80,86	80,24	80,03	80,56	79,57
SE-Inception3	81,02	81,30	80,58	81,53	81,86	81,33
SE-ResNet-18	79,04	79,77	78,87	79,50	79,55	79,42
SE-ResNet-34	79,69	80,27	79,42	79,68	80,13	79,72
SE-ResNet-50	79,94	80,29	79,89	80,16	80,37	79,63
SE-ResNeXt-50	80,74	81,04	80,44	81,29	81,54	81,17
SE-Xception	81,11	81,46	80,54	81,13	81,32	80,99
SE-DenseNet-121	82,36	82,51	82,29	81,96	82,21	81,84

В эксперименте с сокращенной шкалой Келлгрена-Лоуренса, состоящей из четырех классов, наилучшие результаты также были показаны сверточными сетями с архитектурами Inception3 (82,47%) и SE-DenseNet-121 (82,51%). Добавление блоков сжатия и возбуждения к архитектуре Inception не принесло каких-либо улучшений и, напротив, ухудшило точность более чем на 1%. Использование же SE-блоков в архитектуре DenseNet позволило повысить точность

работы более чем на 2%. Результат, превышающий средний результат по качеству работы, также показан архитектурами SE-ResNeXt-50 и Inception-ResNet с точностью 81,54% и 81,32%.

Использование порядковой функции потерь вместо перекрестной энтропии при обучении моделей не принесло какого-либо статистически значимого улучшения. Более того — для архитектуры SE-Inception3 ухудшение качества работы составило 2,39%, при том, что для кросс-энтропии данная архитектура показала один из лучших результатов.

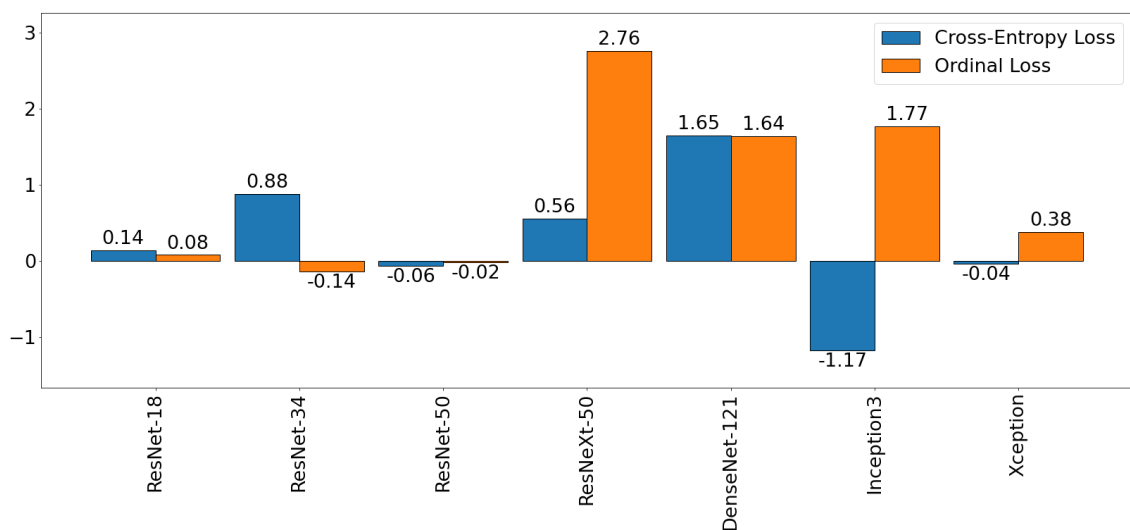


Рисунок 51 — Влияние блоков сжатия и возбуждения на качество работы сверточных сетей традиционных архитектур для сокращенной шкалы Келлгрена-Лоуренса

При этом значения всех показателей оценки классификатора (precision, recall, F1-score) после объединения 0-го и 1-го классов, по сравнению с экспериментом на полной шкале классификации остеоартрита, выросли более чем на 12%. Сравнение подробных метрик качества для топовых моделей (табл. 15) с аналогичными результатами из таблицы 13 показывает, что устранение промежуточного класса 1 из оригинальной шкалы лишь незначительно повлияло на все остальные классы (где-то в лучшую, а где-то в худшую сторону).

На рисунке 52 представлена визуализация матриц ошибок для архитектур SE-DenseNet-121 и Inception, по которым была построена таблица 15.

Таблица 15 — Подробные метрики качества работы классификаторов, показавших наилучшие результаты работы, для каждого из классов (усредненные по трем экспериментам)

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
SE-DenseNet-121						
0	0,87	0,91	0,89	0,85	0,92	0,89
1	0,72	0,67	0,70	0,72	0,65	0,68
2	0,83	0,80	0,81	0,88	0,76	0,81
3	0,89	0,77	0,82	0,84	0,80	0,82
Inception V3						
0	0,85	0,94	0,89	0,83	0,92	0,88
1	0,75	0,62	0,68	0,68	0,61	0,64
2	0,86	0,76	0,81	0,91	0,65	0,75
3	0,81	0,85	0,83	0,81	0,89	0,84

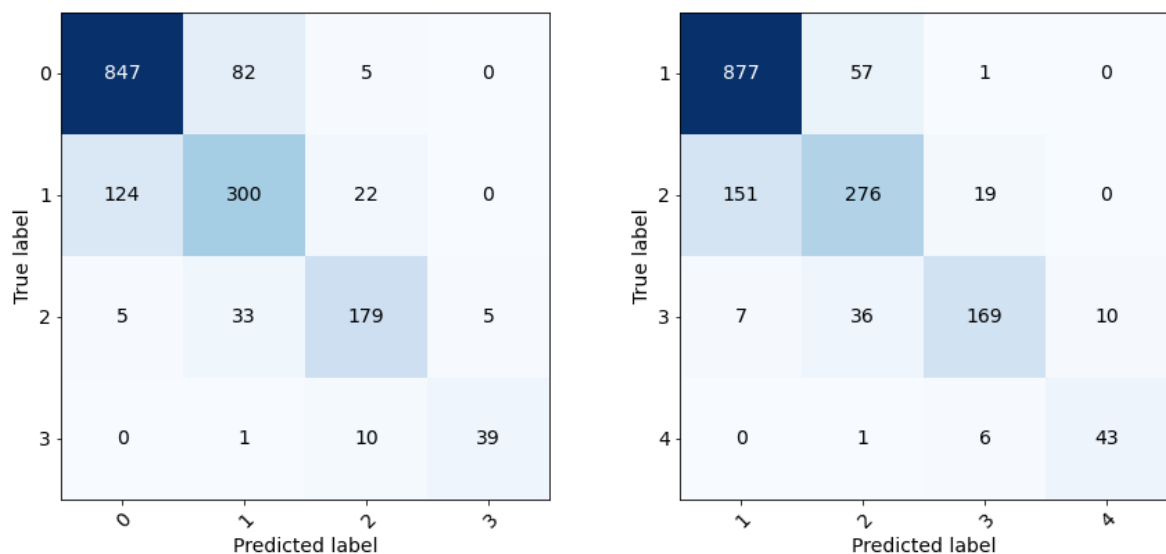


Рисунок 52 — Визуализация матриц ошибок для SE-DenseNet-121 (слева) и Inception (справа)

Бинарная классификация

В экстремальном варианте — к примеру, если рассматривать систему автоматической диагностики ОА как первый этап исследования состояния организма, на котором определяется, болен пациент или нет — классификация по

полной шкале Келлгрена-Лоуренса также может быть избыточна. Достаточно лишь определить, есть ли на представленной рентгенограмме признаки ОА, и в случае положительного ответа передать эту рентгенограмму эксперту для более тщательного изучения. Такая задача, к примеру, решается в работе [88].

В разделе 1.1 отмечалось, что остеоартрит диагностируется на стадиях 2 и выше, а для стадий 0 и 1 пациент считается здоровым. Для полноты эксперимента было проведено исследование поведения классификаторов на задаче бинарной классификации — т.е. его задачей было предсказать, есть ли на поданной на вход рентгенограмме признаки наличия остеоартрита, или они отсутствуют. Для этого, в каждой из моделей последний полносвязный слой с пятью выходами, отвечающий за классификацию степени ОА на изображении, был заменен на слой с двумя выходами.

В качестве модификации датасета, все данные были разбиты на два класса — к первому классу были отнесены изображения, относящиеся к нулевому и первому классам ОА по шкале Келлгрена-Лоуренса, и символизирующие отсутствие ОА, ко второму — все изображения класса 2 и выше по этой шкале, для которых отмечается наличие ОА у пациента. Затем полученный датасет был разбит на обучающую, валидационную и тестовую выборки (см. таблицу 16).

Таблица 16 — Количество изображений в тренировочной, валидационной и тестовой выборках. Non-OA — рентгенограммы здорового коленного сустава, ОА — рентгенограммы, имеющие признаки остеоартрита

Группа	Non-OA	ОА	Всего
Тренировочная выборка	3336	2442	5778
Валидационная выборка	477	349	826
Тестовая выборка	935	721	1656
Процент от общего количества	57,48%	42,52%	100%

Результаты тестирования обученных моделей можно увидеть в таблице 17. По понятным причинам, использование порядковой функции потерь в данном случае лишено смысла, поэтому в таблице представлены результаты только для функции потерь в виде перекрестной энтропии.

Наилучшая усредненная точность классификации — тестирование трех моделей, обученных на разных подмножествах исходных данных с разными начальными значениями генератора случайных чисел, и последующее усреднение их результатов — была получена при помощи архитектур SE-Xception и

Таблица 17 — Среднеклассовые метрики качества работы различных классификаторов для разных функций потерь (усредненные по трем экспериментам)

Архитектура	Cross-Entropy Loss, %		
	Precision	Recall	F1
Antony	84,07	83,74	83,52
ResNet-18	85,98	85,71	85,55
ResNet-34	85,11	84,98	84,88
ResNet-50	85,22	85,00	84,86
ResNeXt	85,32	84,60	84,31
Inception3	86,66	85,57	85,23
Inception-ResNet	86,76	86,23	86,03
Xception	86,32	85,61	85,36
DenseNet-121	85,25	84,76	84,53
SE-Inception3	86,14	85,99	85,88
SE-ResNet-18	84,71	84,58	84,45
SE-ResNet-34	85,74	85,41	85,23
SE-ResNet-50	85,41	85,18	85,02
SE-ResNeXt-50	86,46	86,33	86,24
SE-Xception	86,82	86,59	86,50
SE-DenseNet-121	86,37	86,27	86,18

SE-ResNeXt-50 после добавления к ним блоков сжатия и возбуждения, и составила **86,59%** и **86,33%** соответственно. Сопоставимую точность в 86,27% и 86,23% можно наблюдать в моделях SE-DenseNet-121 и Inception-ResNet. Матрица ошибок и ROC-кривая для SE-Xception показана на рис. 53.

3.3 Использование ансамблей для улучшения точности классификации остеоартрита коленного сустава

У традиционных нейронных сетей существует ряд проблем, которые в некоторых случаях затрудняет их использование в некоторых задачах. К таким проблемам можно отнести медленную сходимость, множественные локальные минимумы, сложность масштабирования и др.

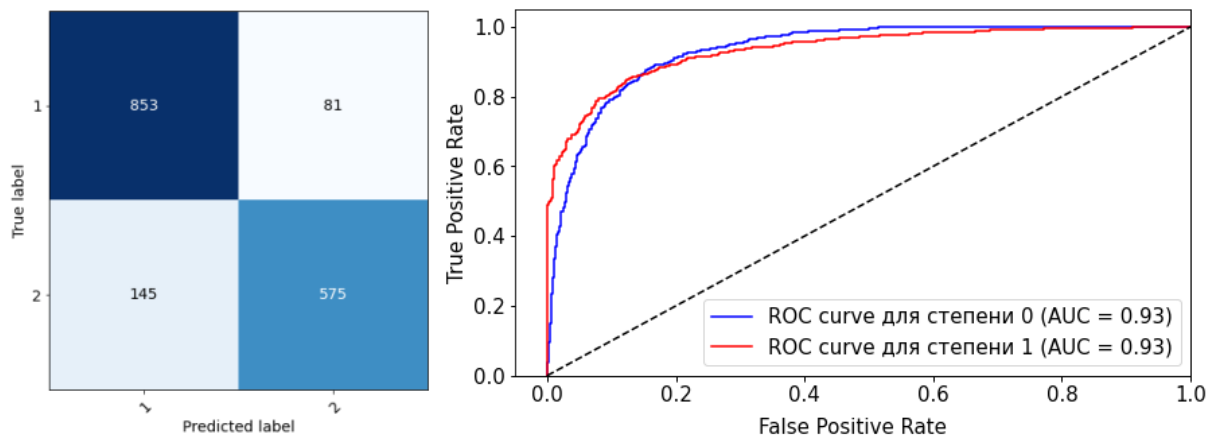


Рисунок 53 — Матрица ошибок (слева) и ROC-кривая (справа) для наилучшего экземпляра SE-DenseNet-121, обученного для решения задачи бинарной классификации. Класс 0 означает отсутствие ОА, класс 1 — наличие признаков ОА на рентгенограмме

Кроме того, точность отдельных моделей зависит от многих факторов — от алгоритма обучения, распределения обучающих данных и т.п. Зачастую, изменение комбинации обучающих данных может привести к значительному улучшению точности работы классификатора. Т.е. в общем случае невозможно точно сказать, изменение каких параметров приведет к повышению точности работы и какая именно версия модели окажется лучшей [123], что добавляет толику неопределенности в процесс обучения.

В разное время предпринимались различные попытки по исследованию методов оптимизации архитектуры и весов нейронных сетей при помощи эволюционных алгоритмов [124], алгоритмов роя частиц [125] и др. В этих работах было показано, что исследуемые подходы действительно позволяют получить неплохие результаты и улучшить производительность нейронных сетей, однако все они обладают высокой вычислительной сложностью и весьма ресурсоемки.

В работе [126], одной из первых, было высказано и доказано предположение о том, что путем объединения результатов нескольких нейронных сетей можно улучшить их производительность. Подобного рода ансамбли обладают более высокой, по сравнению с одиночными моделями, адаптируемостью и могут более качественно работать с неизвестными ранее примерами. В [127] был представлен обзор предыдущих работ по данной тематике и изучены теоретические свойства ансамблей. С этого момента начались активные исследования способов агрегации результатов от нескольких моделей в один, который будет давать более точные предсказания, нежели каждый в отдельности.

3.3.1 Обзор существующих подходов агрегации результатов работы нескольких классификаторов

Ансамблем в машинном обучении называют объединение нескольких «слабых» моделей, обученных отдельно, в одну более сложную модель. Во многих работах было доказано, что использование ансамблей различных классификаторов позволяет улучшить обобщающую способность метода и повысить точность их работы по сравнению с точностью отдельных моделей [128]. Большинство наилучших результатов на крупных конкурсах по машинному обучению (в т.ч. на конкурсе ImageNet) получены именно при помощи ансамблей.

При этом, применение ансамблей для глубокого обучения все еще недостаточно изучено, и в большинстве работ используется вариант с усреднением результатов нескольких моделей, и основные исследования сосредоточены на изучении дизайна архитектур нейронных сетей. Наивное невзвешенное среднее позволяет получить быстрый результат и хорошо работает для случаев моделей с одинаковыми или похожими архитектурами, однако не учитывает специфику данных и уязвимо к плохим моделям.

Существуют различные подходы к объединению нескольких моделей. В качестве основных выделяют бэггинг (**bootstrap aggregating, bagging**), бустинг (**boosting**) и стэкинг (**stacking**) [129; 130].

Bagging. Идея бэггинга [127] состоит в обучении нескольких моделей — как правило, одной архитектуры — на разных подмножествах основного набора данных и последующим агрегированием результатов их предсказания в финальный результат. В зависимости от задачи, в качестве агрегирования обычно используют усреднение ответов нескольких моделей (**weighted averaging**) при решении задачи регрессии или выбор наиболее популярного ответа (**majority voting**) при решении задачи классификации. Усреднение ответов позволяет уменьшить вариабельность (разброс) финальных ответов и сделать итоговую модель более устойчивой к входным данным. Одним из плюсов данного подхода является возможность параллельного обучения различных классификаторов, т.к. они не зависят друг от друга и могут быть натренированы независимо.

Одним из наглядных примеров бэггинга является случайный лес, состоящий из множества решающих деревьев. Точность отдельных деревьев невысока,

однако агрегация результатов работы большого количества деревьев на некоторых классах задач позволяет получить неплохую точность.

Невзвешенное среднее (или *unweighted averaging*) является одним из наиболее общих подходов к созданию ансамбля нейронных сетей и заключается в использовании невзвешенного среднего предсказаний нескольких независимых моделей для получения финального результата или финального распределения вероятностей. Усреднение выходных значений нейронных сетей позволяет уменьшить дисперсию, причем в некоторых случаях (особенно тогда, когда модели ансамбля слабо скоррелированы между собой) довольно значительно. Вероятность для класса k для подобного способа составления ансамбля можно выразить следующим образом:

$$s[k] = \frac{\exp \left[\sum_{m=1}^M s_m[k] \right]}{\sum_{j=1}^K \exp \left[\sum_{m=1}^M s_m[j] \right]}, \quad (4)$$

где M — это количество моделей в ансамбле, K — общее количество классов, а $s_m[k]$ — индивидуальный выход соответствующей модели перед слоем Softmax (т.н. ненормализованное распределение вероятностей). При этом, можно усреднять как непосредственно выход нейронной сети, так и вероятности, полученные после применения операции softmax.

Голосование большинством (или *majority voting*) схоже с невзвешенным средним, однако в данном случае вместо усреднения выходных вероятностей подсчитывается количество голосов за каждый класс от участвующих в ансамбле моделей, и в качестве финального ответа выбирается тот класс, который получил наибольшее количество голосов.

В отличие от невзвешенного среднего, голосование большинством менее чувствительно к выходу одиночной модели, однако в случае, если в ансамбле используются несколько похожих моделей, финальный результат может зависеть от них. Кроме того, т.к. используются только финальные метки классов, часть информации может быть потеряна.

Взвешенное среднее (или *weighted averaging*) заключается в составлении результирующего выходного вектора вероятностей как линейной комбинации выходов из участвующих в ансамбле моделей:

$$s[k] = \sum_{m=1}^M a_m s_m[k], \quad (5)$$

где M — это количество моделей в ансамбле, а $s_m[k]$ — индивидуальный выход соответствующей модели перед слоем Softmax для класса k , а вектор весов $a_i, i = 1..M$ получен обучением некоего мета-классификатора.

Супер-классификатор (или Super Learner [123]) является одной из разновидностей взвешенного среднего и использует k -блочную кросс-валидацию (**k-fold cross-validation**). Алгоритм состоит из следующих шагов:

1. Разбиение тренировочной выборки V на k непересекающихся подмножеств $V = \bigcup_{i=1}^k V_i$.
2. Выбор m моделей, называемых базовыми, для создания ансамбля.
3. Для каждой базовой модели выполняется кросс-валидация на полученном разбиении $V_i, i = 1..k$, полученные оценки сохраняются, и затем модель обучается на полном тренировочном множестве V .
4. На данных, полученных в результате кросс-валидации для всех моделей, обучается некий мета-классификатор, который позволяет получить коэффициенты для линейной комбинации базовых классификаторов ансамбля.
5. Для оценки тестового множества используется линейная комбинация базовых моделей, обученных на полном тренировочном множестве, с коэффициентами, вычисленными мета-классификатором по результатам кросс-валидации.

В работе [127] отмечается, что с некоторого момента, с увеличением количества моделей в ансамбле, уменьшения ошибки предсказания ансамблем не происходит, т.е. в случае бэггинга нет прямой зависимости между количеством моделей в ансамбле и уменьшением ошибки классификации.

Boosting. Бустинг [131] относится к последовательным методам комбинирования моделей. Это означает, что отдельные модели теперь не могут быть натренированы параллельно, что значительно увеличивает время обучения. В отличие от бэггинга, который позволяет сделать результаты предсказания более устойчивыми, бустинг направлен на последовательное, итеративное улучшение точности каждого следующего слабого классификатора.

«Сильная» модель по-прежнему берется как взвешенная сумма «слабых» моделей. При этом, при обучении слабых классификаторов, каждая следующая «слабая» модель обычно обучается так, чтобы давать лучший результат на том наборе данных, на котором предыдущий классификатор делал ошибки.

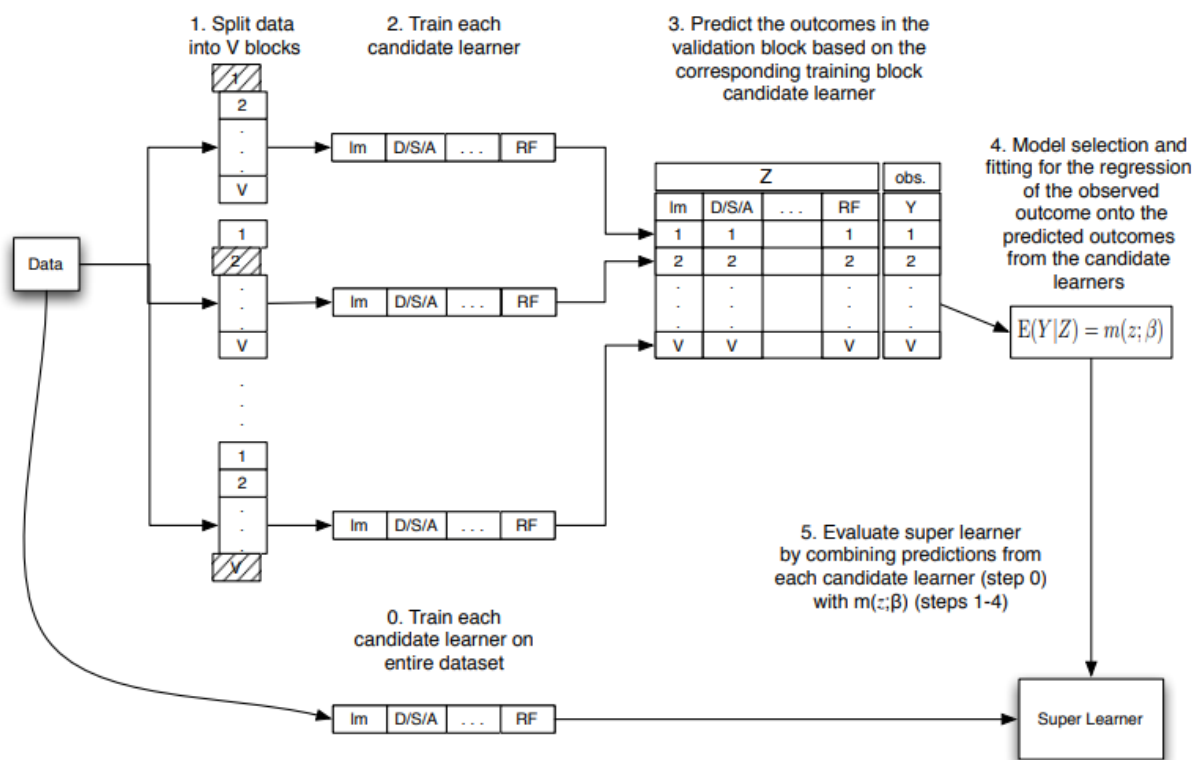


Рисунок 54 — Диаграмма супер-классификатора из работы [123]

Отличительной особенностью бустинга является то, что при выборе тренировочной выборки соответствующие наборы данных выбираются не случайно, как в предыдущих случаях, а в зависимости от предыдущих результатов слабых классификаторов. Также, если бустинг дает эффект в улучшении производительности, то этот эффект, как правило, лучше, нежели использование бэггинга, однако в случае неудачи использование бустинга может привести к ухудшению производительности системы.

Stacking. Бэггинг и бустинг, как правило, используются над набором моделей с одинаковой архитектурой, но обученных на разных наборах данных, т. е. модели различаются лишь весами. Стекинг отличается от двух этих подходов тем, что в нем могут использоваться модели разной архитектуры. Кроме того, вместо простого взвешенного среднего слабых моделей при данном подходе дополнительно обучается некий мета-классификатор, который комбинирует результаты предсказания простых моделей. Данный мета-классификатор принимает на вход результаты предсказания этих простых моделей и учится наилучшим образом давать финальный результат на их основе. Помимо обычного стекинга, существует также и многоуровневый стекинг — в этом случае мета-классификатор обучается на базе других мета-классификаторов, в свою очередь обученных на некоторых наборах слабых моделей.

Данный подход требует наличия более объемных датасетов, чем в предыдущих случаях, т.к. обучать мета-классификатор необходимо на данных, которые не участвовали в обучении слабых классификаторов. Именно поэтому в работе подобный подход к созданию ансамблей не используется.

3.3.2 Эксперимент

В диссертации в качестве ансамбля использовался бэггинг трех моделей, обученных на одном и том же наборе данных, но с разными начальными значениями для генератора случайных чисел (21, 42, и 84), что позволило обучать одну и ту же модель на разных комбинациях исходных данных. При этом, для каждого случая использовались модели, показавшие наилучший результат на валидационной выборке. В качестве функций агрегации выбрано два варианта — невзвешенное среднее и голосование большинством [130].

Напомним, что для голосования большинством в качестве финального ответа выбирается тот, за который проголосовало большинство из имеющихся в ансамбле моделей. Для вычисления невзвешенного среднего предсказания всех моделей, используемых в ансамбле, суммировались и пропускались через слой *Softmax*. Вероятность степени остеоартрита K прогрессирования ОА по шкале KL можно выразить следующим образом:

$$P(y = k|x) = \frac{\exp \left[\sum_{m=1}^M P_m(y = k|x) \right]}{\sum_{j=1}^K \exp \left[\sum_{m=1}^M P_m(y = j|x) \right]}, \quad (6)$$

где $M = 3$ это количество моделей в ансамбле, $K = 5$ — количество классов, а $P_m(y = k|x)$ — индивидуальный выход соответствующей модели перед слоем *Softmax* (т.н. ненормализованное распределение вероятностей).

Классификация степени остеоартрита по полной шкале Келлгрена-Лоуренса

При тестировании ансамблей сетей, обученных классифицировать остеоартрит по полной шкале Келлгрена-Лоуренса, для каждой архитектуры использовались три модели, полученные при разных начальных значениях генератора случайных чисел, что позволило обучать одни и те же модели на разных комбинациях изображений из исходных наборов данных.

Из-за ограниченного размера датасета были рассмотрены такие способы комбинирования моделей, как невзвешенное среднее (UA) и голосование большинством (MV).

Таблица 18 — Среднеклассовые метрики качества работы различных классификаторов для разных функций потерь (усредненные по трем экспериментам)

Метод	UA, %		MV, %	
	CE	ORD	CE	ORD
ResNet-18	66,73	68,12	66,43	68,06
ResNet-34	68,12	67,81	68,24	66,24
ResNet-50	68,42	68,72	68,00	68,24
ResNeXt	67,69	68,72	69,20	67,45
Inception3	72,10	71,74	71,62	71,74
Inception-ResNetV2	71,98	69,75	71,07	69,02
Xception	70,95	71,44	69,44	70,95
DenseNet-121	69,14	69,20	68,30	68,00
SE-Inception3	71,68	70,65	70,89	69,69
SE-ResNet-18	67,63	67,09	66,73	66,24
SE-ResNet-34	68,96	69,26	68,12	67,75
SE-ResNet-50	69,38	66,67	69,08	68,84
SE-ResNeXt-50	71,92	71,98	72,28	71,86
SE-Xception	71,44	70,95	70,65	70,35
SE-DenseNet-121	71,44	70,29	70,90	70,41

Наилучшие результаты использования ансамблей получены для архитектуры **SE-ResNeXt-50** и составили **72,28%** для голосования большинством и функции потерь в виде перекрестной энтропии. Сопоставимы результаты с точ-

ностью **72,10%** можно наблюдать для архитектуры **Inception3** для функции агрегации в виде невзвешенного среднего.

Для подавляющего большинства всех архитектур использование ансамблей вместо одиночных моделей позволило повысить качество работы более чем на 1%, для отдельных моделей (Inception3, SE-Xception, DenseNet-121 и др.) этот показатель превысил 2%.

Классификация степени остеоартрита по сокращенной шкале Келлгрена-Лоуренса

Таблица 19 — Среднеклассовые метрики качества работы различных классификаторов (4 класса) для разных функций потерь (усредненные по трем экспериментам)

Метод	Unweighted Averaging,%		Majority Voting,%	
	CE	ORD	CE	ORD
ResNet-18	80,43	80,92	80,92	80,07
ResNet-34	80,74	80,98	80,25	81,58
ResNet-50	81,64	81,58	81,64	81,34
ResNeXt	81,70	81,16	81,40	80,74
Inception3	83,57	81,52	83,64	81,09
Inception-ResNetV2	82,61	82,55	81,40	82,73
Xception	82,79	82,01	82,85	82,91
DenseNet-121	81,89	81,22	82,00	81,10
SE-Inception3	83,03	83,21	82,37	82,73
SE-ResNet-18	80,74	80,68	80,92	80,50
SE-ResNet-34	81,52	81,46	82,25	82,37
SE-ResNet-50	82,07	82,67	81,04	82,31
SE-ResNeXt-50	82,43	82,37	82,37	82,19
SE-Xception	82,55	82,91	82,49	82,55
SE-DenseNet-121	84,66	83,09	84,66	83,15

В отличие от конфигурации с полной шкалой Келлгрена-Лоуренса классификации остеоартрита, при использовании ансамблей для сокращенной шкалы

лидерами по точности работы остались те же самые модели, которые были таковыми в одиночном варианте. Наилучший результат здесь был получен для моделей **SE-DenseNet-121** и **Inception3** с точностью в **84,66%** и **83,64%**, при этом улучшение составило более 2%. Матрица ошибок и ROC-кривая для ансамбля моделей SE-DenseNet-121 представлена на рис. 55, подробные метрики классификации — в таблице 19.

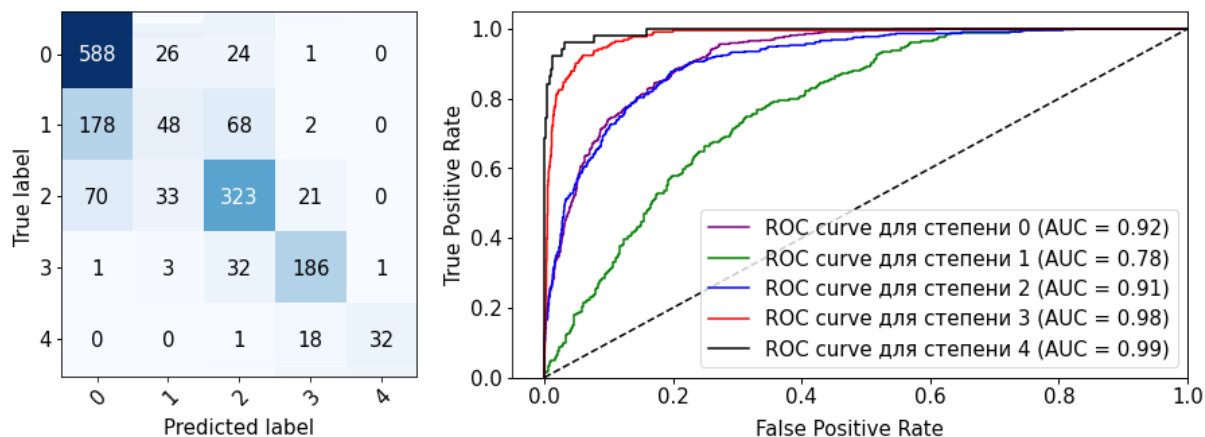


Рисунок 55 — Матрица ошибок (слева) и ROC-кривая (справа) для ансамбля сетей DenseNet-121

Можно отметить, что для сокращенной шкалы результаты более стабильные, и в целом коррелируют с результатами одиночных моделей. Вероятно, это связано с устранением неоднозначности после объединения в один класс 0-го и 1-го классов полной шкалы Келлгрена-Лоуренса — в финальных метриках по результатам тестирования моделей эта неоднозначность явно выделялась по сравнению с результатами на остальных классах.

Определение наличия признаков остеоартрита на рентгенограмме

В данном разделе приводятся результаты экспериментов с обучением моделей предсказывать факт наличия или отсутствия остеоартрита на изображении. Отсутствие остеоартрита диагностируется для 0-го и 1-го классов остеоартрита по шкале Келлгрена-Лоуренса, наличие остеоартрита разной степени тяжести фиксируется для оставшихся трех классов. Результаты экспериментов представлены в таблице 20. В скобках указана величина улучшения,

полученная при использовании нескольких моделей вместо наилучших результатов одиночных моделей соответствующих архитектур.

Таблица 20 — Результаты работы различных архитектур на задаче определения факта наличия остеоартрита на рентгенограмме — по каждому из пунктов показаны усредненные данные для нескольких моделей, обученных с разным случайным начальным значением

Метод	UA,%	MV,%
ResNet-18	86,65 (0,94)	86,78 (1,07)
ResNet-34	85,87 (0,89)	86,11 (1,13)
ResNet-50	86,29 (1,29)	86,35 (1,35)
ResNeXt-50	81,28 (-3,32)	73,79 (-10,81)
Inception3	85,87 (0,30)	86,17 (0,60)
Inception-ResNetV2	86,90 (0,67)	86,96 (0,73)
Xception	86,65 (1,04)	87,44 (1,83)
DenseNet-121	85,14 (0,38)	85,27 (0,51)
SE-Inception3	87,20 (1,21)	87,26 (1,27)
SE-ResNet-18	85,75 (1,17)	85,81 (1,23)
SE-ResNet-34	86,41 (1,00)	86,11 (0,70)
SE-ResNet-50	86,84 (1,66)	86,35 (1,17)
SE-ResNeXt-50	86,65 (0,32)	86,71 (0,38)
SE-Xception	87,50 (0,91)	86,90 (0,31)
SE-DenseNet-121	87,32 (1,05)	86,96 (0,69)

Наилучший результат был получен для архитектуры **SE-Xception** и составил в итоге **87,50%** точности классификации в случае невзвешенного среднего. Улучшение по сравнению со средним результатом одиночных моделей составило 0,91%. Матрица ошибок и ROC-кривая для наилучшего ансамбля из моделей SE-Xception представлены на рис. 56, подробные метрики классификации для каждого класса можно найти в таблице 21.

В целом, использование ансамблей позволило улучшить финальный результат для каждой из моделей на 1%, за исключением архитектуры ResNeXt-50, где деградация составила 3,32% для невзвешенного среднего и более 10% для случая голосования большинством. Кроме того, для варианта ResNeXt-50 с блоками сжатия и возбуждения использование ансамблей для обоих видов функции агрегации составило в среднем символические 0,35%.

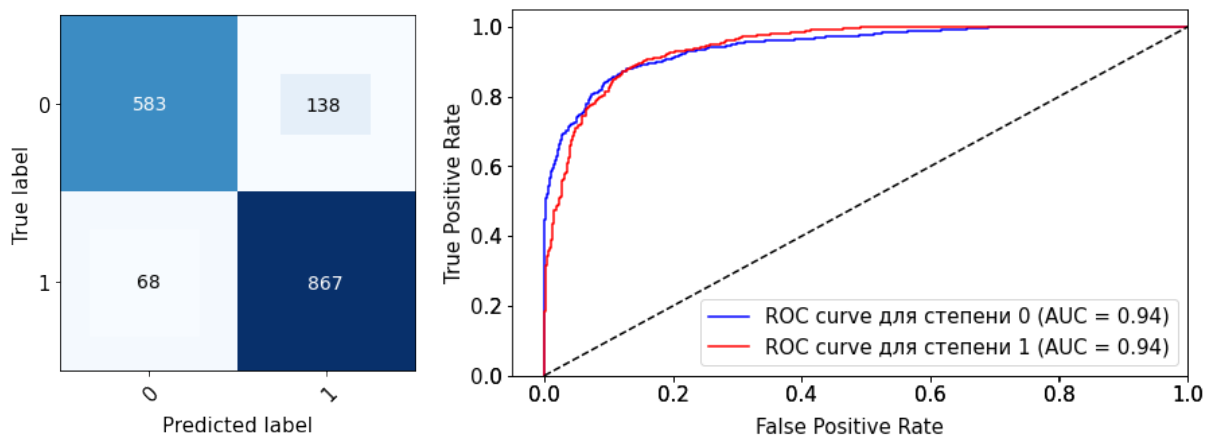


Рисунок 56 — Матрица ошибок (слева) и ROC-кривая (справа) для ансамбля сетей SE-Xception для задачи бинарной классификации

Таблица 21 — Метрики результатов ансамбля сетей SE-Xception для задачи бинарной классификации

	Precision	Recall	F_1
Non-OA	0,90	0,81	0,85
OA	0,86	0,93	0,89
Среднее	0,88	0,88	0,88

3.4 Обсуждение результатов

В ходе экспериментов было проведено обучение 16 различных архитектур (10 оригинальных и 6 разновидностей архитектуры ResNet и SE-ResNet различной глубины), каждая из которых включала в себя версии для трех шкал оценки остеоартрита — оригинальной пятиклассовой шкалы Келлгрена-Лоуренса, сокращенной четырехклассовой шкалы и экстремального случая бинарной классификации наличия признаков ОА на рентгенограмме. Для каждой из моделей было проведено три процедуры обучения с разными начальными данными генератора случайных чисел — значениями 21, 42 и 84, усредненные финальные показатели эффективности представлены в результирующих таблицах с метриками качества.

Обнаружено, что наилучший результат среди оригинальных архитектур для всех видов шкалы оценки ОА получен при помощи сверточных нейронных сетей с архитектурой Inception3. При рассмотрении тех же самых архитектур с блоками сжатия и возбуждения (SE-блоками), сети с архитектурами

SE-DenseNet-121, SE-ResNeXt-50 и SE-Xception смогли превзойти результаты IncetionV3 на всех шкалах оценки остеоартрита при использовании ансамблей. При этом, использование SE-блоков для архитектуры IncetionV3 **ухудшило** точность ее работы более чем на 1%.

В таблице 22 представлено изменение количества обучаемых параметров сверточных сетей различных архитектур при добавлении блоков сжатия и возбуждения (SE-блоков). Наибольшее изменение отмечается в архитектурах DenseNet-121(43,82%), ResNeXt-50(11,00%) и ResNet-50(10,76%) ввиду особенностей расположения этих блоков. Кроме того, повышение точности работы явно коррелирует с количеством SE-блоков, т.к. наилучшее повышение качества работы после добавления блоков зафиксировано для архитектур DenseNet-121 и ResNeXt-50.

Таблица 22 — Изменение количества параметров модели при добавлении SE-блоков в известные архитектуры

Архитектура	Стандарт	С SE-блоками	Изменение
ResNet-18	11,18M	11,27M	+ 0,09 (0,008%)
ResNet-34	21,29M	21,44M	+ 0,15 (0,007%)
ResNet-50	23,52M	26,05M	+ 2,53 (10,76%)
ResNeXt-50	22,99M	25,52M	+ 2,53 (11,00%)
Inception3	21,80M	22,94M	+ 1,14 (5,23%)
Xception	20,82M	21,56M	+ 0,74 (3,55%)
DenseNet-121	6,96M	10,01M	+ 3,05 (43,82%)

Изменение точности используемых архитектур при использовании порядковой функции потерь показано на рис. 57 — отмечается, что в большинстве случаев использование подобной функции потерь не принесло желаемого эффекта а, напротив, ухудшило финальную точность моделей, за исключением варианта с архитектурой Xception для полной шкалы Келлгрена-Лоуренса.

На рис. 58 в виде гистограммы изображено влияние изменения способов оценки остеоартрита путем сокращения шкалы Келлгрена-Лоуренса на финальную точность классификации. Видно, что объединение нулевого и первого классов ОА шкалы Келлгрена-Лоуренса в один класс (отсутствие остеоартрита) позволило более чем на 10% повысить эффективность работы исследуемых классификаторов, а переход к бинарной классификации повышает точность

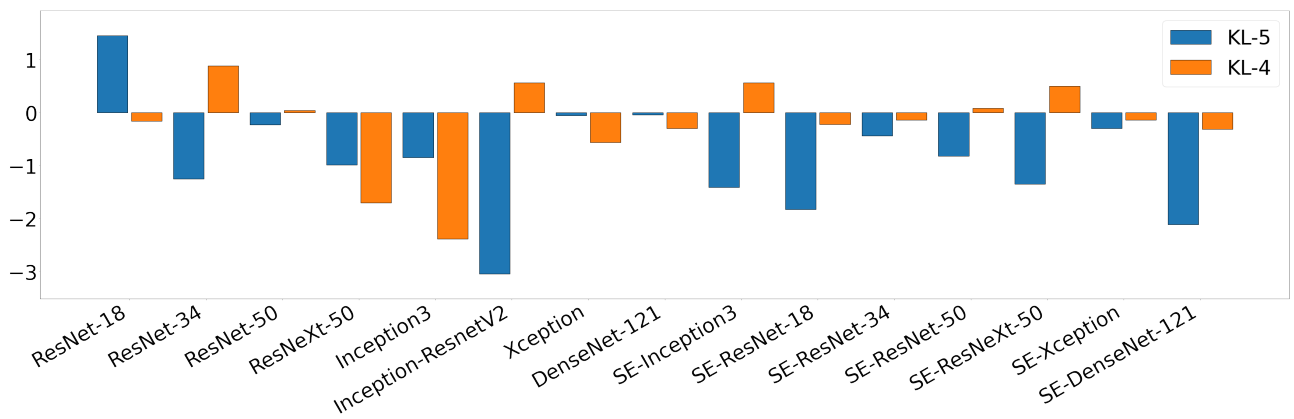


Рисунок 57 — Влияние использования порядковой функции потерь при обучении на финальное качество работы сверточных сетей

почти на 20%. Это подтверждает гипотезу о неоднозначности шкалы Келлгрена-Лоуренса, которая неоднократно озвучивалась в предыдущих исследованиях [4].

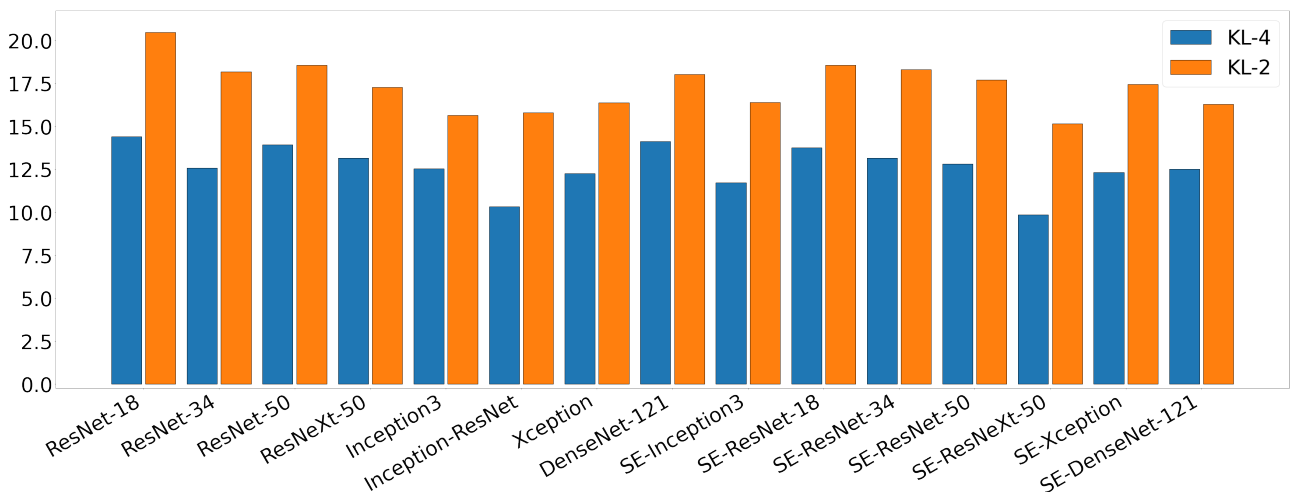


Рисунок 58 — Изменение точности классификации ОА коленного сустава при помощи известных архитектур сверточных нейронных сетей при сокращении шкалы Келлгрена-Лоуренса до четырехклассовой (KL-4) и двухклассовой (KL-2)

На рис. 59 представлена гистограмма, показывающая улучшение точности работы озвученных в работе классификаторов на базе сверточных нейронных сетей при использовании вместо одиночных моделей ансамбля таких моделей. В качестве метода агрегации для сравнения использовался метод невзвешенного среднего, в качестве функции потерь — перекрестная энтропия.

Отмечаем, что в большинстве случаев при использовании ансамблей удалось повысить точность более чем на 1%. Наибольшее повышение при этом зафиксировано для случая полной шкалы Келлгрена-Лоуренса, и составило 2,41% для архитектуры DenseNet-121. Наименьшее улучшение (в среднем) получено для случая бинарной классификации.

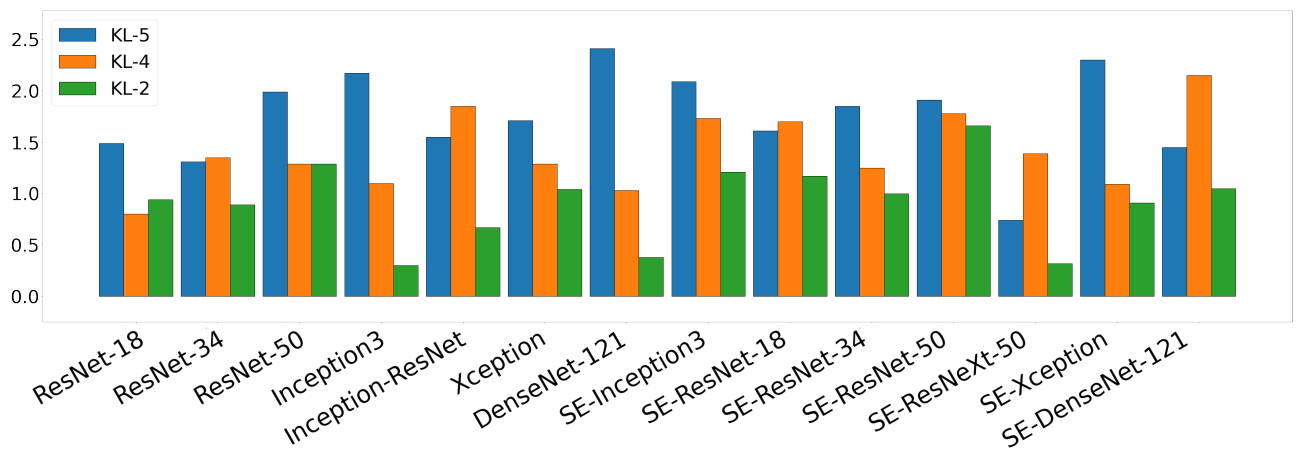


Рисунок 59 — Улучшение точности классификации при использовании ансамблей

Единственный случай ухудшения на 3,32% при использовании ансамблей имеет место быть для архитектуры ResNeXt-50 для случая бинарной классификации. Учитывая, что для всех остальных случаев использование ансамблей архитектурой ResNeXt-50 позволило получить улучшение финальной точности, можно сказать, что данное ухудшение в целом не меняет общую картину в отношении использования ансамблей.

Таким образом можно сделать вывод, что методы, использующие ансамбли моделей, являются мощным инструментом, позволяющим получить более качественные результаты по сравнению с использованием отдельных классификаторов. Достигается это за счет того, что методы усредняют ошибки каждого базового классификатора и уменьшают влияние случайностей. Также, использование нескольких моделей позволяет уменьшить дисперсию, поскольку несколько разных моделей, исходящих из разных гипотез, имеют больше шансов прийти к правильному результату, чем одна отдельно взятая.

В сводной таблице 23 представлена информация о точности классификации из всех упомянутых в диссертации работ, включая результаты, полученные в проведенных исследованиях. В колонке *Данные* указано общее количество суставов, которые были использованы при финальном тестировании метода, значение *OAI* означает использование для этих целей части датасета The Osteoarthritis Initiative (табл. 4). За исключением нескольких случаев наилучшие результаты показаны моделями, которые были обучены в рамках диссертации. Однако эти несколько случаев требуют отдельных пояснений.

В работах [50; 80; 84; 96] заявлена достаточно высокая точность классификации для полной шкалы Келлгрена-Лоуренса, однако тестирование прово-

Таблица 23 — Сравнение методов автоматической диагностики остеоартрита для разных методов оценки

Метод	KL-5	KL-4	KL-2	Данные
WND-CHARM [56; 82]	38%	-	-	2243
Random Forest over stat. [80]	87,5%	-	-	200
SVM over HOG [84]	90,0%	-	-	616
SVM + k-средних [50]	>95,0%	-	-	30
SSM + случ. лес по бок. пр. [87]	47,9%	-	78,9%	300
SSM + случ. лес по фронт. пр. [88]	-	-	84,0%	500
Полносвязная нейронная сеть [91]	66,6%	-	-	<100
CNN, Antony et all. [65; 92]	64,6%	-	-	OAI
DenseNet + антропометр. данные [93]	-	77,2%	-	OAI
VGG-16 + LSTM [94]	75,28%	-	-	475
VGG-16 + Attention + GAP [95]	64,3%	-	-	OAI
Faster R-CNN [96]	76,5%	-	-	550
VGG-19 + Ordinal Loss [68]	67,7%	-	-	OAI
Сиамская CNN [97]	66,71%	-	-	OAI
Усреднение двух ResNet 18, 34, 50 [99]	66,68%	-	-	OAI
SE-ResNeXt-50 + град. бустинг [100]	70,0%	-	-	OAI
SE-ResNeXt-50 [100]	68,0%	-	-	OAI
VGG-16 + U-Net + случ. лес [102]	64,48%	-	-	OAI
VGG-16 + U-Net + случ. лес [102]	75,86%	-	-	MOST
VGG-16 [102]	71,93%	-	-	MOST
Inception-ResNet-v2 + походка [103]	64,7%	-	-	728
SE-ResNeXt-50 (наш метод)	71,18%	81,54%	86,33%	OAI
SE-DenseNet-121 (наш метод)	69,99%	82,51%	86,27%	OAI
3 × SE-ResNeXt-50 (наш метод)	72,28%	82,43%	86,71%	OAI
3 × SE-DenseNet-121 (наш метод)	71,44%	84,66%	87,32%	OAI
3 × SE-Xception (наш метод)	71,44%	82,55%	87,50%	OAI

дилось на довольно ограниченном наборе изображений — 30 и 200 для первых двух работ, 616 для третьей и 550 для четвертой. Это не позволяет в полной мере оценить качество исследуемых алгоритмов классификации.

В работе [94] при использовании полной шкалы Келлгрена-Лоуренса также заявлена точность, превышающая точность моделей, полученных в рамках

написания диссертации. Для каждого из классов точность распознавания составляет $75 \pm 1,5$, что отличается от аналогичных результатов как в большинстве остальных работ, так и от результатов тестирования моделей в проведенных экспериментах. Во всех остальных работах точность классификации для класса 1 существенно ниже точности классификации для всех остальных классов. Например, по таблице 13 можно увидеть, что она более чем в два раза ниже точности для остальных классов. Подобный результат наблюдается и во всех остальных работах и объясняется нечеткими критериями отличия стадий 0 и 1. Это может свидетельствовать об ошибках в проведении экспериментов в работе [94] или отличающейся системе оценивания качества классификации. Кроме того, данный результат был показан на выборке из 475 изображений, что также недостаточно для получения объективных результатов.

Еще один превосходящий результат в 75,86% описан в работе [102] на датасете MOST. При этом на датасете OAI результат аналогичного метода составил 64,48%. Таким образом, имело место различие в используемых датасетах для обучения и тестирования, что не позволяет установить однозначное соответствие между результатами.

Для случаев сокращенной шкалы Келлгрена-Лоуренса и бинарной классификации предложенные модели однозначно опередили по точности классификации все предыдущие методы с результатами 84,66% и 87,5% соответственно для ансамблей из трех моделей SE-DenseNet-121 и SE-ResNeXt-50.

В открытом доступе отсутствуют исследования, направленные на сравнение результатов работы автоматических алгоритмов и результатов диагностики специалистом-человеком, однако можно судить о данном показателе по косвенным признакам. Так, в датасете CHECK [13] оценка остеоартрита по атласам OARSI для каждой рентгенограммы выполнялась 5 экспертами, и нередко результаты этих экспертов различались друг от друга. В работе [102] приводится средняя численная оценка эффективности диагностики остеоартрита по шкале Келлгрена-Лоуренса несколькими экспертами на датасете MOST [12] при помощи коэффициента k (взвешенная каппа Коэна [40]), которая равна **0,748**. В таблице 24 приводятся подсчитанные значения коэффициента k для моделей, полученных в диссертации и показавших наилучшие результаты.

Озвученную эффективность диагностики остеоартрита человеком в 0,748 можно косвенно сравнивать со значением коэффициента для ансамбля из 3-х моделей SE-ResNeXt-50, где $k = 0,849$. Таким образом, можно сделать вывод, что

Таблица 24 — Значение коэффициента k для лучших моделей, полученных в рамках диссертации, по всем используемым шкалам оценки ОА из таблицы 23

Метод	KL-5	KL-4	KL-2
$3 \times \text{SE-ResNeXt-50}$	0,849	-	-
$3 \times \text{SE-DenseNet-121}$	-	0,873	-
$3 \times \text{SE-Xception}$	-	-	0,744

эффективность диагностики предложенным в диссертации алгоритмом на 10,1% превосходит эффективность диагностики ОА человеком на схожем по структуре и количеству примеров наборе данных.

3.5 Выводы

В соответствии с поставленными задачами в третьей главе были получены следующие результаты.

1. Проведено обучение сверточных нейронных сетей современных архитектур, адаптированных для классификации остеоартрита по полной, сокращенной шкалам Келлгрена-Лоуренса и для случая бинарной классификации.
2. Проведена оценка результатов работы обученных моделей на тестовом наборе данных и использованы разные метрики для сравнения этих результатов с результатами из уже опубликованных работ.
3. Исследованы различные подходы к созданию ансамблей нейронных сетей, а также осуществлена программная реализация и тестирование ансамблей на базе изученных ранее моделей, как для задачи классификации степени ОА по шкале KL, так и для задачи бинарной классификации наличия признаков остеоартрита на рентгенограмме.
4. Проведен анализ полученных результатов.

Результаты данной главы опубликованы в [141; 148; 142].

Глава 4. Разработка метода детектирования контуров костей на медицинских рентгенографических изображениях

В главе описывается новый алгоритм выделения контуров костей на медицинских рентгенограммах. Предложенный алгоритм акцентирует внимание на корректном выделении границ, которые позволят точно выделить контуры интересующих нас объектов [145]. В рамках исследования были определены критерии, позволяющие вычислить численную характеристику качества выделения контуров на изображении [140; 143]. Предложен метод устранения разрывов границ на изображении, а также метод отслеживания контуров на результирующем бинарном изображении и дополнительная энергия для активных контуров [146], учитывающие особенности рентгенографических изображений конечностей.

Детектирование контуров можно рассматривать как этап вычисления количественных характеристик автоматизированной диагностики ОА после этапа автоматической классификации, в котором мы получаем стадию заболевания без подробных деталей. Полученные результаты могут быть использованы для определения типа представленных на рентгенограмме объектов, характера и степени сужения межсуставного пространства, определения параллельности мышечковых частей бедренной и большеберцовой костей, вычисления углов между костями и т.п.

Использование детектирования контуров, наряду с этапом автоматической локализации и классификации степени остеоартрита, позволяет получить достаточно информации для постановки диагноза или для существенного упрощения данной процедуры, тем самым позволяя снизить нагрузку на специалиста и уменьшить фактор субъективности.

4.1 Разработка метода выделения контуров костей на рентгенограмме

Предлагаемый в работе метод для поиска контуров костей коленного сустава использует комбинацию различных техник обработки изображений. Помимо применения фильтров размытия, порогового разделения и метода активных кон-

туров на разных этапах также используется некоторая информация об основных характеристиках объектов исследования (примерный цвет на рентгенограмме, особенности распределения яркости и т.д.). Подробное пошаговое описание метода представлено ниже.

4.1.1 Сглаживание и фильтрация шума

Распространенной практикой в области обработки изображений является применение размытия к входному изображению перед его непосредственной обработкой для избавления от разного рода шума [52]. Одним из распространённых методов предварительного сглаживания изображения является размытие фильтром Гаусса, который можно записать следующим образом:

$$GF[I]_p = \sum_{q \in S} G_\sigma(\|p - q\|) I_q, \quad \forall p \in I, \quad (7)$$

где S — окрестность пикселя p , а $G_\sigma(x)$ — двухмерное ядро гауссиана с параметром σ :

$$G_\sigma(x) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2}{2\sigma^2}\right).$$

Хорошо видно, что вес для пикселя p определяется гауссианом G_σ с аргументом $\|p - q\|$, т.е. влияние на итоговый цвет пикселя p соседние пиксели оказывают только на основе расстояния между ними — чем дальше пиксель от центра, тем меньший коэффициент к нему применяется. Из-за этого в процессе размытия информация о границах на изображении может теряться, что является весьма существенным недостатком данного вида размытия.

В рамках предложенного алгоритма предварительное сглаживание исходного изображения предлагается выполнять при помощи билатеральной фильтрации [132; 133]. Билатеральный фильтр является нелинейным шумоподавляющим фильтром, который сохраняет границы. Интенсивность каждого пикселя изображения по-прежнему рассчитывается как взвешенное среднее его соседей в определенной окрестности, однако важной особенностью является то,

что в весовых коэффициентах учитывается не только расстояние между пикселями, но и разность между интенсивностью пикселей. Фильтр может быть записан в форме

$$BF[I]_p = \frac{1}{W_p} \sum_{q \in S} G_{\sigma_s}(\|p - q\|) G_{\sigma_r}(|I_p - I_q|) I_q, \quad \forall p \in I, \quad (8)$$

где W_p является коэффициентом нормализации, которые обращает сумму весовых коэффициентов в единицу:

$$W_p = \sum_{q \in S} G_{\sigma_s}(\|p - q\|) G_{\sigma_r}(|I_p - I_q|).$$

Равенство 8 определяет нормированное взвешенное среднее яркости пикселей изображения I в окрестности точки p . G_{σ_s} определяет влияние на вес соседнего пикселя расстояния между пикселями p и q , а G_{σ_r} определяет похожесть интенсивностей пикселей. На рис. 60 наглядно продемонстрировано действие билатерального фильтра на центральный пиксель изображения.

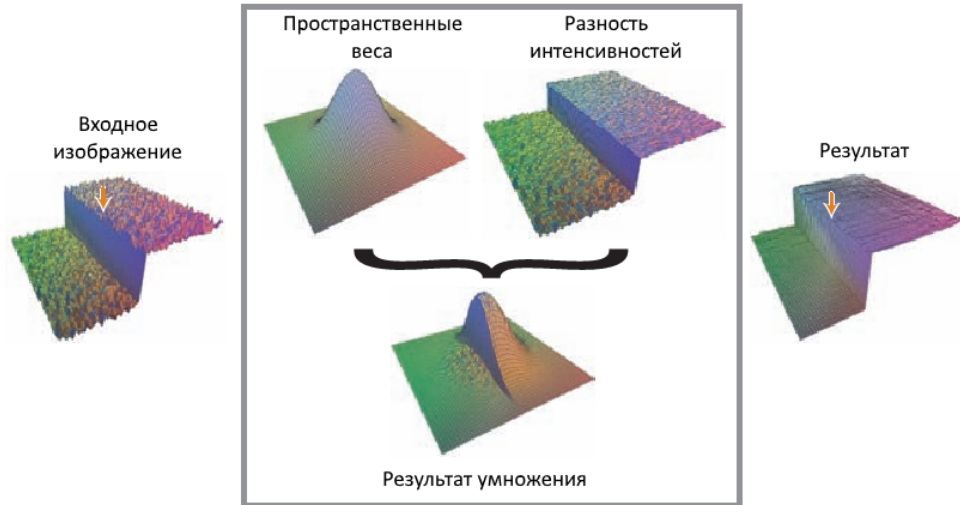


Рисунок 60 — Иллюстрация билатерального фильтра из работы [134]. Весовые коэффициенты применяются к пикселю под стрелкой

4.1.2 Вычисление вспомогательной маски объектов

В дальнейшем для вспомогательных целей нам потребуется маска, определяющая примерное положение объектов на изображении (в нашем случае

объектами являются кости). Учитывая, что пиксели, принадлежащие кости, на рентгенограмме светлого цвета, получим эту маску при помощи применения простого порогового разделения методом Отсу [135] к исходному изображению после билатеральной фильтрации (рис. 61, а).

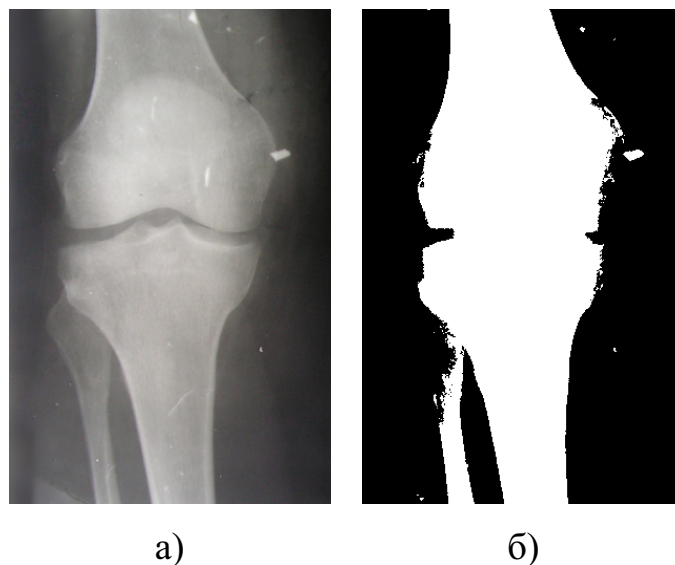


Рисунок 61 — Пример бинаризации изображения с порогом, вычисленным методом Отсу, для получения примерной маски объектов

Чтобы устранить единичные белые пиксели, «дыры» внутри костей и прочие недостатки грубого порогового разделения, к бинарной версии дополнительно применяется операция морфологического расширения с небольшим радиусом. Обозначим результат бинаризации после применения морфологических операций как I^{bin} . Пример такого бинарного изображения приведен на рис. 61, б.

4.1.3 Вычисление градиента изображения и поиск «сильных» фрагментов границы объектов

Помимо бинарной версии, на базе сглаженного изображения вычисляется его градиент. В классическом варианте для этого может быть использован оператор Собеля, в работе же в качестве основного оператора вычисления градиента использовался оператор Кирша [136], который хорошо зарекомендовал себя в области обработки медицинских изображений (рис. 62, а). Через $|\nabla I|$ и I^{dir} будем обозначать значение модуля и направления этого градиента соответственно.

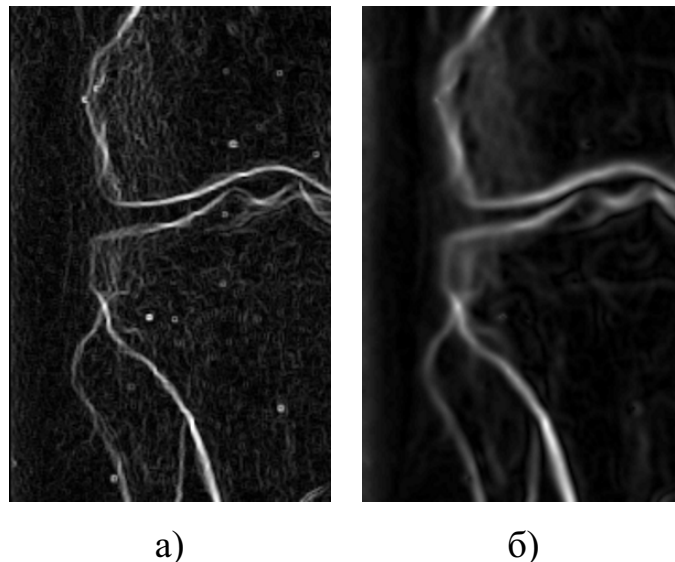


Рисунок 62 — Иллюстрация к этапу вычисления градиента: значение модуля градиента сглаженной версии исходного изображения $|\nabla I|$, вычисленное при помощи оператора Кирша (а) для участка изображения и векторное поле потока градиента для того же участка (б)

Градиент изображения используется для поиска краев, однако минусом стандартных градиентных операторов (и оператора Кирша в том числе) применительно к обработке изображений является то, что область действия этого поля невелика — большие значения принимаются лишь вблизи границ объектов на изображении, а на остальных участках значения модуля градиента такого поля близки к нулю.

Чтобы устранить этот недостаток, в работе дополнительно используется поле векторного потока градиента (gradient vector flow, GVF [137]). Значения данного поля вычисляются итеративно на базе стандартного градиентного оператора при помощи уравнения диффузии. Обозначим модуль и направление карты GVF как $|\mathbf{v}|$ и $\mathbf{v}^{dir}$. Через $G_1 = |\mathbf{v}| * |\nabla I|$ обозначим изображение, являющееся результатом поэлементного перемножения модулей градиента и модулей GVF (рис. 63, а). Можно заметить, что, по сравнению с $|\nabla I|$, результирующее изображение менее зашумлено в областях, не относящихся к границе, и в то же время имеет более сильный отклик в пикселях, близких к границе объектов.

Для изображения G_1 вычисляется верхнее значение порога бинаризации $\overline{T}$, которое будет в дальнейшем использовано для поиска наилучшего порога. Алгоритм вычисления $\overline{T}$ состоит из двух шагов.

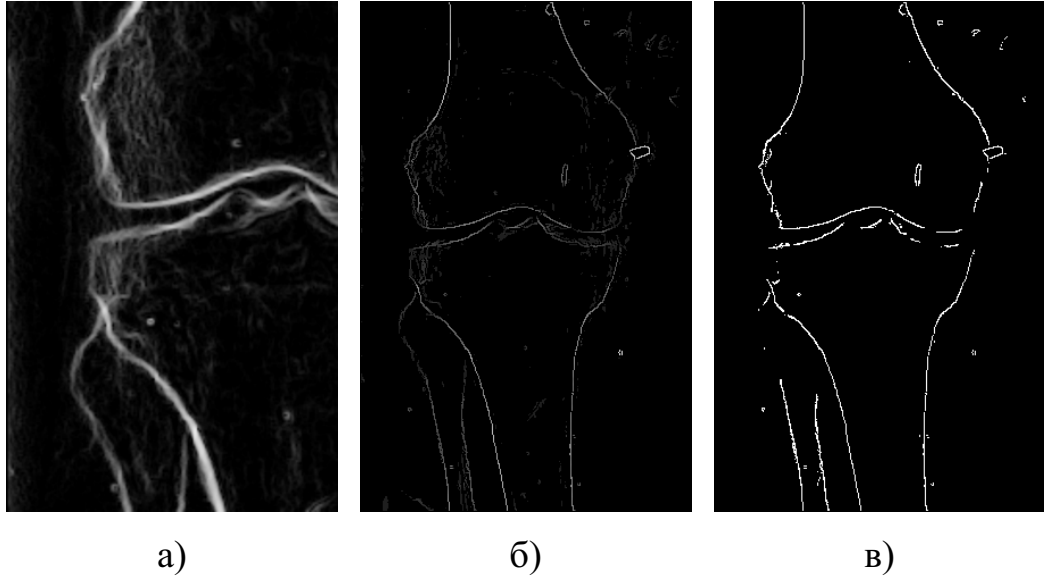


Рисунок 63 — Иллюстрация этапа формирования «сильных» фрагментов границ: результат поэлементного перемножения модулей градиента и GVF (а), результат применения подавления не максимумов к изображению G_1 (б) и результат бинаризации G_2 с некоторым порогом из множества $\hat{T}$ (в)

1. Определяется модуль градиента яркости для каждого пикселя изображения:

$$\hat{G}(x, y) = \max(|G_{1,x}(x, y)|, |G_{1,y}(x, y)|).$$

Для вычисления производных по направлению $G_{1,x}$, $G_{1,y}$ используется дискретный аналог оператора дифференцирования:

$$G_{1,x}(x, y) = G_1(x + 1, y) - G_1(x - 1, y),$$

$$G_{1,y}(x, y) = G_1(x, y + 1) - G_1(x, y - 1).$$

2. Вычисляется порог по формуле:

$$\bar{T} = \frac{\sum_{y=1}^N \sum_{x=1}^M G_1(x, y) \cdot \hat{G}(x, y)}{\sum_{y=1}^N \sum_{x=1}^M \hat{G}(x, y)}. \quad (9)$$

Также, используя поле направлений $\mathbf{v}^{dir}$, к G_1 применяется процедура подавления не максимумов. Величины направлений округляются до ближайших 45° , что соответствует 8-связной окрестности пикселя. Эта операция аналогична той, которая используется в методе Кэнни выделения границ на изображении [52]. После этого получим изображение G_2 , с которым продолжим работать на следующем этапе.

На основе верхнего ограничения $\overline{T}$, как разбиение интервала $(0, \overline{T}]$, формируется множество порогов $\hat{T} = \{T_i\}_{i=1, \overline{l}}$. Данное множество будет использовано для поиска оптимального порога бинаризации изображения G_2 .

4.1.4 Устранение разрывов и объединение фрагментов границ в контуры

Через G^{bin} обозначим результат бинаризации G_2 с некоторым порогом. Это изображение состоит из фрагментов границ (рис. 63, в). В работе предложено два алгоритма связывания фрагментов, принадлежащих одному контуру, для устранения разрывов между ними.

Точки разрыва

Сначала введем понятие точки разрыва границы. Напомним, что на данном этапе работа выполняется над бинарным изображением, в котором белым цветом обозначены точки границы объектов, а черным — все остальные. Далее, говоря о точках, будем подразумевать именно точки границы.

Обозначим через $U(p)$ окрестность точки p — множество смежных с ней пикселей границы на изображении (т.е. точек, в которых $G^{bin}(p_i) = 1$), при этом будем рассматривать случай 8-связной окрестности пикселя ($|U(p)| \leq 8$). Через $\rho(p_1, p_2)$ будем обозначать Евклидово расстояние между точками p_1 и p_2 . Определим три вида точек, которые будем считать точками, с которых начинается разрыв границы (контура) объекта:

- **Первый тип.** Точка, для которой $|U(p)| \leq 1$ (рис. 64, а).
- **Второй тип.** Точка, у которой есть две смежные точки, принадлежащие контуру, причем стоящие рядом на одной вертикали/горизонтали (рис. 64, б):

$$U(p) = \{p_1, p_2\} :$$

$$\rho(p_1, p_2) = 1, x_1 = x_2 \vee y_1 = y_2.$$

- **Третий тип.** Точка, у которой есть три смежные точки, принадлежащие контуру, которые являются смежными друг другу, т.е. все три не лежат одновременно на одной вертикали или горизонтали (рис. 64, в):

$$U(p) = \{p_1, p_2, p_3\} : \rho(p_1, p_2) = 1, \rho(p_1, p_3) = 1, x_1 = x_2, y_1 = y_3.$$

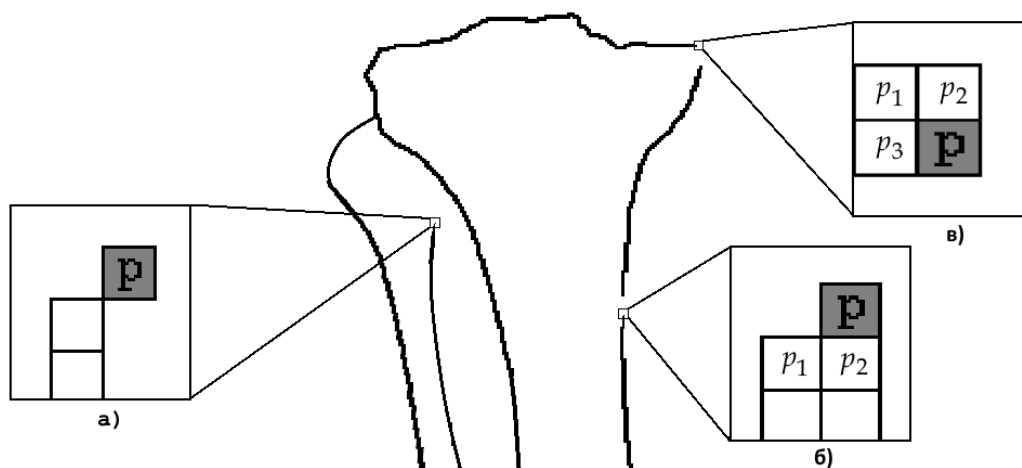


Рисунок 64 — Примеры точек разрыва: точки разрыва первого (а), второго (б) и третьего (в) типов

В ходе экспериментов было обнаружено, что данного набора точек разрыва достаточно для решения проблемы связывания границ при помощи разработанных алгоритмов устранения разрывов.

Алгоритм состоит из двух этапов. Сначала производится устранение разрывов между парами крайних пикселей соседних фрагментов. При этом используется несколько попыток с различными ограничениями в виде максимально допустимой длины разрыва между точками и минимально возможной «стоимости» набора устраняющих разрыв точек. Вторым этапом обрабатываются точки разрыва, не имеющие пары. Такие разрывы часто возникают в местах соединения нескольких костей, когда вторая из точек, принадлежащая разрыву, из-за особенностей рентгенографической проекции, на изображении не видна. При этом также используется несколько итераций с различными ограничениями. Пример поэтапного устранения разрывов представлен на рис. 65.

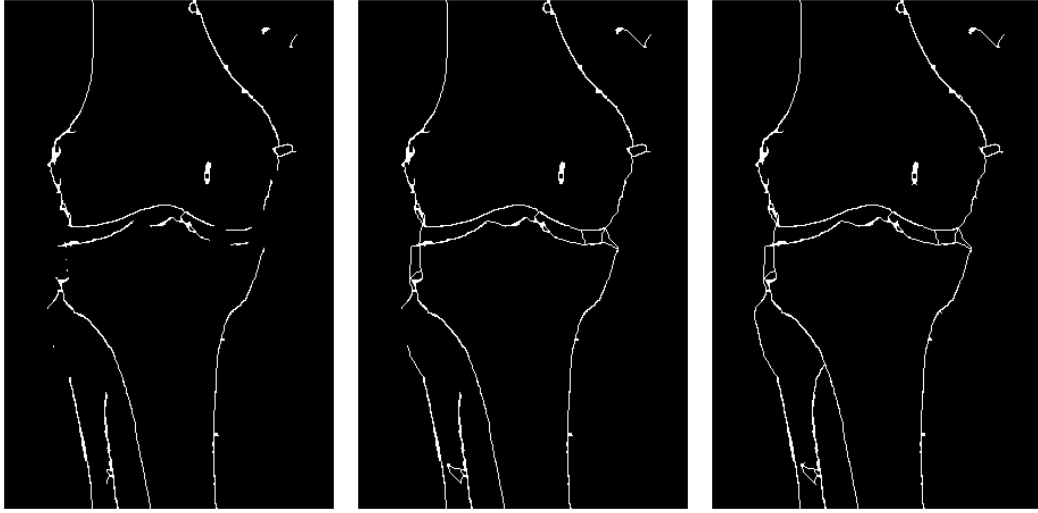


Рисунок 65 — Пример поэтапного устранения разрывов границ: слева направо показано состояние изображения для 0, 25 и 55 итерации предложенного алгоритма соответственно

Основной алгоритм устранения разрывов

Условие принадлежности пары точек одному разрыву. Для ограничения количества комбинаций пар точек среди всех имеющихся на изображении точек разрыва используется условие, опирающееся на вероятную кривизну участка разрыва границы. Пусть p и q — точки разрыва, r — одна из смежных с p точек, вектор $\bar{u}$ сонаправлен с вектором $\overline{p-r}$, d — константа, определяющая возможный диапазон отклонения точки q от точки p в направлении $\bar{u}$ (рис. 66). Зная направление вектора $\bar{u}$ и значение d , можно определить вектора $\bar{v}$ и $\bar{w}$, которые ограничивают область поиска точки разрыва, соответствующей точке p :

$$v_x = u_x \cos(-d) - u_y \sin(-d),$$

$$v_y = u_x \sin(-d) + u_y \cos(-d),$$

$$\omega_y = u_x \cos d - u_y \sin d,$$

$$\omega_x = u_x \sin d + u_y \cos d.$$

В качестве условия, при котором будет произведена попытка устранения разрыва между точками p и q , будем рассматривать ограничения

$$\omega_x t_y - \omega_y t_x \geq 0, \quad t_x v_y - t_y v_x < 0, \quad (10)$$

где $\bar{t} = q - p$. Это условие означает, что точка q должна находиться между ограничивающими векторами $\bar{v}$ и $\bar{w}$, проведенными из точки p .

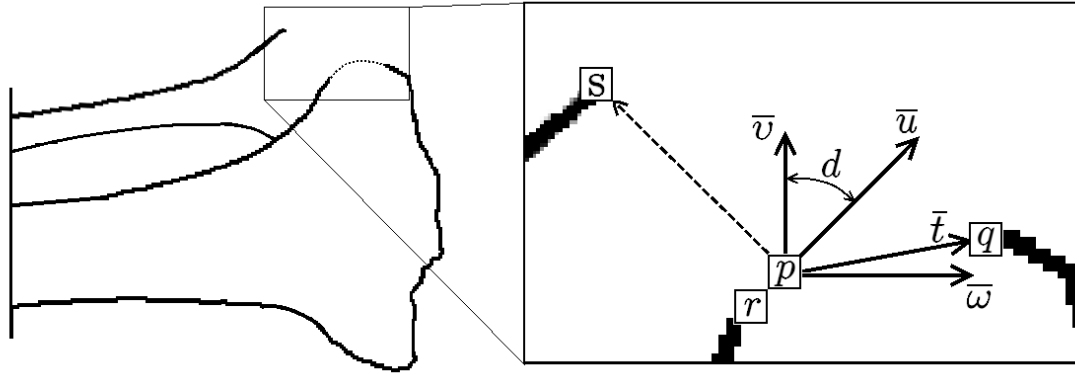


Рисунок 66 — Иллюстрация к условию существования разрыва между точками p и q . В данном случае условие выполняется — q лежит между $\bar{v}$ и $\bar{w}$

Устранение разрыва. Предположим, что выбраны две точки, между которыми будет устраняться разрыв — условия для выбора точек было озвучено в предыдущем разделе. Устранять разрыв будем при помощи алгоритма поиска пути на двумерной сетке — алгоритма A^* , который является расширением алгоритма Дейкстры. Будем использовать следующие эвристики:

- стоимость прохождения через точку p — значение модуля градиента изображения в этой точке $|\nabla I(p)|$, взятое со знаком «минус»;
- стоимость перехода из одной точки в другую — евклидово расстояние между точками и модуль разности между модулями градиента;
- при выборе следующей точки при прочих равных приоритет отдается той, которая ближе к цели (ко второй точке разрыва).

Предварительно производится масштабирование значений $|\nabla I|$ до некоторого диапазона. Чем больше диапазон возможных значений модуля градиента, тем более точно путь будет повторять границу исследуемого объекта. Но на изображении в некоторых случаях существуют участки, которые содержат близкие границы двух различных объектов. На таких участках в процессе поиска алгоритм может «перескочить» на границу другого объекта из-за особенностей градиента. Таким образом, нужно соблюдать осторожность при выборе коэффициента масштабирования. Хорошие результаты на практике давало отображение $|\nabla I|$ в диапазон $[0, 10]$.

В результате работы алгоритма мы получим некоторый набор точек, являющийся оптимальным путем между двумя точками для соответствующих

входных данных и выбранных эвристик. Обозначим этот набор $\Gamma = \{\gamma_i\}_{i=1}^n$. Обозначим через ξ среднее значение модуля градиента во всем изображении, Φ — стоимость всего пути, которая подсчитывается при обратной трассировке как сумма модулей градиента в каждой его точке:

$$\xi = \frac{1}{NM} \sum_{x=1}^M \sum_{y=1}^N |\nabla I(x, y)|, \quad (11)$$

$$\Phi = \sum_{i=1}^n |\nabla I(\gamma_i)|. \quad (12)$$

Рассматривать Γ в качестве точек, с помощью которых будет устранен разрыв, будем в том случае, если выполняется условие

$$\Phi > n \cdot \xi \cdot \theta, \quad (13)$$

где n — количество точек Γ , а θ является параметром алгоритма (чем больше θ — тем более жесткие требования предъявляются к Γ). Итак, предположим, что условие (13) выполняется.

Если Γ не пересекает уже существующую границу на изображении G^{bin} (рис. 67), то фиксируем Γ как набор точек, который устраняет разрыв между текущими точками разрыва, и отмечаем это на G^{bin} . Напомним, что этот алгоритм не рассматривает пути, размеры которых больше некоторого числа K ($0 < L \leq K$), являющегося параметром алгоритма.

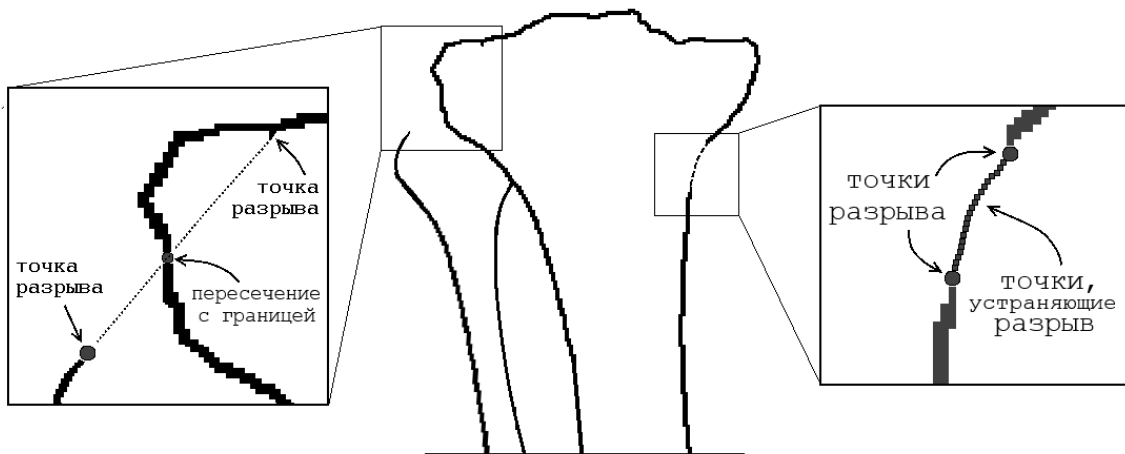


Рисунок 67 — Нарушение условия о не пересечении существующей границы (слева) и пример устранения разрыва (справа)

Дополнительный алгоритм устранения разрывов

Предположим, что выбрана точка, которая является разрывной. Из этой точки начинает формироваться кривая, «растекаясь» по градиенту. Следующая точка кривой выбирается в направлении, сонаправленном с перпендикуляром к направлению градиента в предыдущей точке. Удлинение кривой прекращается, как только она встретит на своем пути точку с указанными свойствами. К примеру, условиями остановки могут быть встретившаяся точка, принадлежащая границе объекта (15) или выход за границу изображения (14).

$$P1(x, y) = \begin{cases} true, & (x, y) \notin [0, N) \times [0, M), \\ false, & else. \end{cases} \quad (14)$$

$$P2(x, y) = \begin{cases} true, & G_2^{bin}(x, y) = 1, \\ false, & else. \end{cases} \quad (15)$$

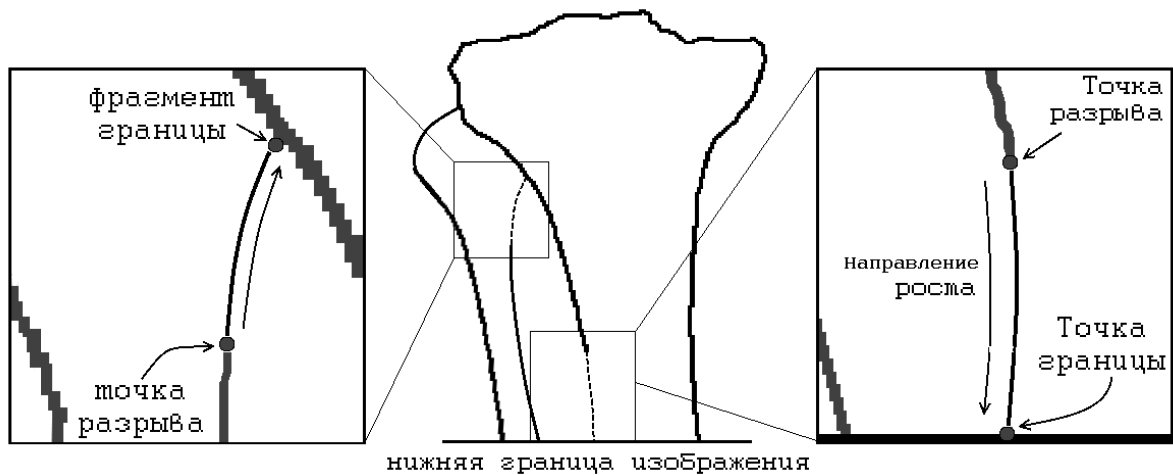


Рисунок 68 — Пример устранения разрыва для условия 15 (слева) и устранение разрыва для условия 14 (справа)

Как и в предыдущем алгоритме, считаем, что эта кривая устраняет разрыв, если выполняется (13). Примеры устранения разрывов по указанным условиям приведены на рис. 68.

Порядок применения алгоритмов устранения разрывов

Для получения хороших результатов необходимо комбинировать основной и вспомогательный алгоритмы устранения разрывов.

В таблице 25 представлен порядок применения алгоритмов, а также ограничения и коэффициенты, которые были использованы при тестировании разработанного комплекса алгоритмов устранения разрывов на бинарном изображении с размерами в пределах 500×500 пикселей.

Таблица 25 — Порядок применения алгоритмов, тестовые значения ограничений и коэффициентов

	Алгоритм	K	θ
1	Морфологическое замыкание изображения	-	-
Отображение значений модуля градиента в диапазон $[0, 10]$			
2	Основной алгоритм	8	1,5
3	Основной алгоритм	16	1,25
4	Основной алгоритм	32	1,5
Отображение значений модуля градиента в диапазон $[0, 255]$			
5	Дополнительный алгоритм (14)	16	1,0
6	Дополнительный алгоритм (14)	48	1,75
7	Дополнительный алгоритм (15)	32	1,75

Вначале несколько раз применяется основной алгоритм с различной допустимой длиной K и коэффициентами θ . После применения основного алгоритма на изображении могли остаться точки, которые являются разрывными, но парной разрывной точки не имеют. Для таких случаев применяется дополнительный алгоритм поочередно с двумя разными условиями — ищем границу изображения и границу объекта на изображении.

4.1.5 Метод отслеживания контура на изображении

Выбор ключевых точек. После завершения процедуры устранения разрывов мы получаем бинарное изображение $\overline{G}$, которое содержит замкнутые области.

Каждая такая область соответствует искомым объектам на изображении, чьи контуры необходимо найти. Ниже представлен разработанный в рамках исследования метод определения ключевых точек контура для замкнутой области, показавший при тестировании наилучшие результаты среди всех рассматриваемых подходов, и метод восстановления контуров по этим ключевым точкам.

Сначала необходимо определить точку, наиболее близко расположенную к центру масс области, и при этом принадлежащую этой области. Обозначим такую точку (x_c, y_c) . Для определения одной ключевой точки в некотором направлении проводится луч с началом в точке (x_c, y_c) , и определяется пересечение этого луча с границей области. При этом в качестве ключевой точки нужно выбирать не первую встретившуюся точку границы, а точку, наиболее удаленную от (x_c, y_c) . Предположим, что (p_i, p_{i+1}) — две последовательные точки такого луча, при этом $\bar{G}(p_i) = 1$, т.е. p_i — точка границы. Точка p_i выбирается в качестве одной из ключевых точек рассматриваемой области только в том случае, когда p_{i+1} принадлежит другой области. Т.е. метки области, для которой мы пытаемся найти контур, и области, которой принадлежит точка p_{i+1} , различны.

Необходимый набор ключевых точек определяется путем сканирования изображения лучом из точки (x_c, y_c) в направлениях из диапазона $[0, 2\pi]$ с заданным шагом. Схематично пример поиска ключевых точек показан на рис. 69. При этом порядок ключевых точек в контуре определяется величиной угла, под которым луч эту точку обнаружил.

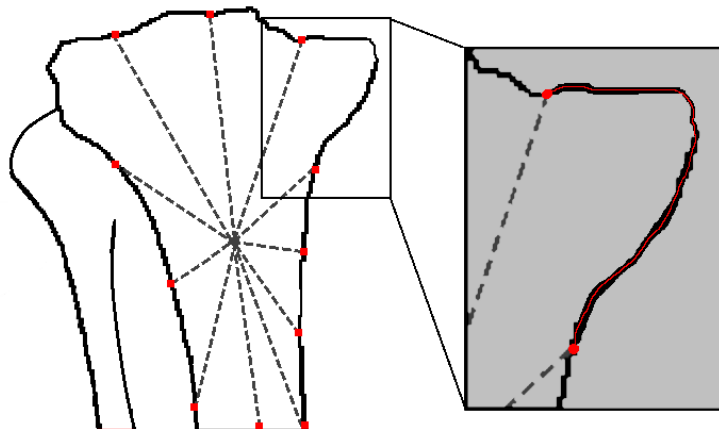


Рисунок 69 — Общая идея алгоритма отслеживания контура

Формирование контура. После того, как необходимый набор ключевых точек найден, с помощью упомянутого выше алгоритма A^* происходит поиск пути между последовательными парами этих точек. Поле градиента, используемое

для оценки стоимости прохождения через точку, должно быть отображено в более широкий диапазон значений, нежели в случае использования этого поля для устранения разрывов (в эксперименте использовался диапазон $[0, 255]$). В этом случае расстояние между точками оказывает на формируемый путь гораздо меньше влияния, и путь более точно описывает участок контура объекта между двумя последовательными ключевыми точками.

Данный подход позволяет корректно обрабатывать участки бинарного изображения, которые лишь приблизительно указывают границу объекта. Стандартные методы, не учитывающие значения градиента изображения и работающие только с бинарной его версией, не способны правильно обработать подобные проблемные участки (рис. 70).

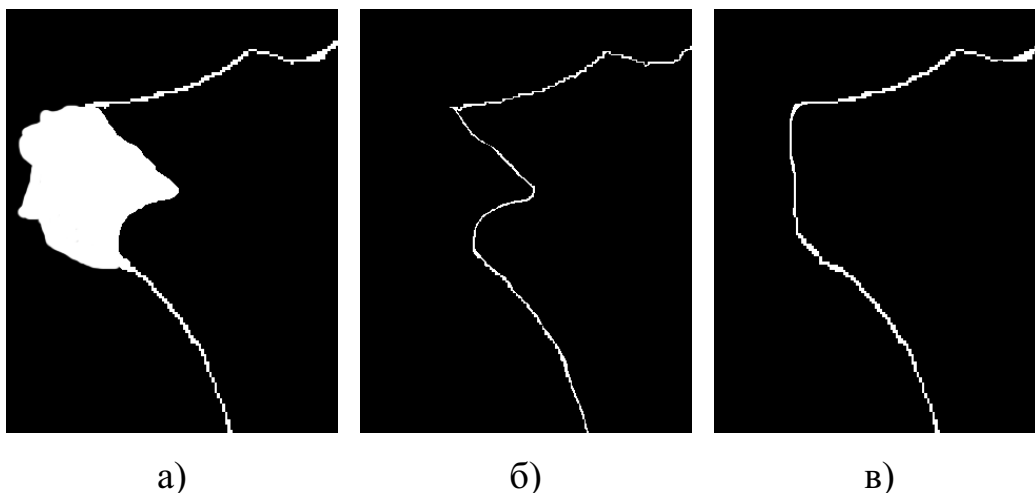


Рисунок 70 — Иллюстрация преимущества предлагаемого метода на примере участка с возможностью неоднозначной трактовки границы: пример проблемного участка бинарного изображения (а), фрагмент контура, найденный одним из стандартных алгоритмов отслеживания контуров — «алгоритмом жука» (б) и фрагмент контура, найденный предложенным методом с использованием карты градиента (в)

4.1.6 Уточнение и сглаживание финальных контуров

Контуры, найденные при помощи описанного выше алгоритма, довольно точно описывают границу объекта. Однако, при формировании этих контуров алгоритм опирался исключительно на градиент изображения, игнорируя

другие возможные факторы, которые могли оказать на них влияние. Чтобы уменьшить вероятность некорректного отслеживания отдельных участков, для уточнения полученных контуров предлагается использовать *метод активных контуров* [110]. Данный алгоритм учитывает различные факторы при модификации контуров, что при верном начальном положении шаблона (а в нашем случае это выполняется) позволяет повысить точность детектирования. Для сглаживания результирующих контуров используется модель, описанная в работе [138]. В рамках работы предложена специальная функция энергии, которая позволяет улучшить качество работы активных контуров применительно к поставленной задаче.

Стандартная модель. Активный контур — это некоторый набор точек $V = (v_1, \dots, v_n)$, которые могут изменять свое положение под действием некоторых сил, итеративно приближаясь к истинной границе объекта, при помощи минимизации энергии контура

$$E(V) = \sum_{i=0}^n E_{int}(v_i) + E_{ext}(v_i),$$

где E_{int} — внутренняя энергия, зависящая от формы контура, E_{ext} — внешняя энергия, зависящая от параметров изображения (таких как интенсивность пикселей, градиент и т.п.). Подробное описание энергий, составляющих E_{int} и E_{ext} в выбранном методе, можно найти в [138].

Для минимизации энергии контура в работе используется *жадный алгоритм*, краткое описание которого можно найти в работе [138], а более полное — в [110]. Общая его идея состоит в том, что энергии $E_{int}(v_i)$ и $E_{ext}(v_i)$ для каждой точки контура рассматриваются как квадратные матрицы, и значение в центре такой матрицы соответствует значению соответствующей энергии контура в точке v_i (рис. 71). Остальные значения матриц соответствуют энергиям контура в окружающих v_i точках. Каждая вершина v_i может перейти в любую соседнюю точку v'_i с целью минимизации общей энергии контура. На каждой итерации процесса деформации контура каждая из его точек старается минимизировать его новую энергию при помощи смены позиции на более оптимальную.

Предлагаемая энергия. В работе предлагается новая составляющая внешней энергии — E_{dir} , которая на практике позволила достичь лучшей точности работы

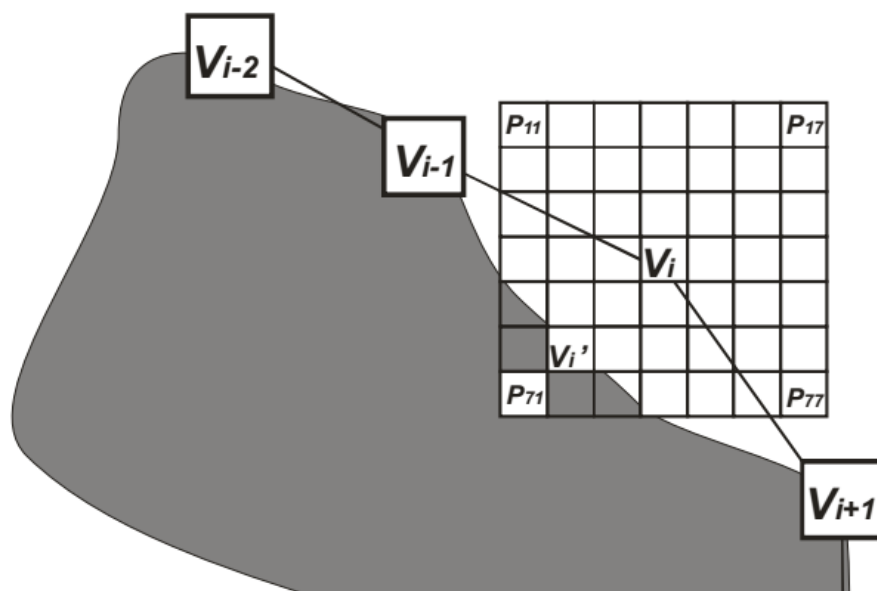


Рисунок 71 — Иллюстрация жадного алгоритма в методе активных контуров из работы [138]: в качестве текущей точки рассматривается v_i , v_i' — изменение положения для минимизации общей энергии всего контура

метода. Эта энергия позволяет предотвратить смещение точек контура вдоль границы объекта в процессе минимизации энергии и поощряет движение точки в направлении изменения яркости (рис. 72). Стрелками обозначено направление градиента на представленном участке изображения.

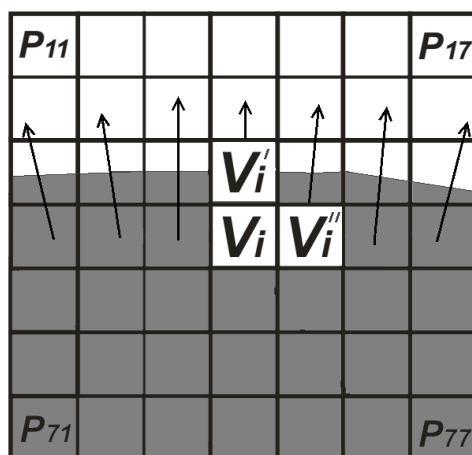


Рисунок 72 — Иллюстрация действия новой энергии E_{dir} . Стрелками схематично обозначено направление градиента на представленном фрагменте изображения — в данном случае $E_{dir}(v_i') > E_{dir}(v_i'')$

Энергия является своего рода альтернативой и дополнением к распирающей энергии контура, которая отвечает за расширение контура или его сужение к центру, оптимизируя пути этого расширения или сужения. Таким образом,

финальная внешняя энергия для одной точки контура будет вычисляться по формуле:

$$E_{ext}(v_i) = E_{std}(v_i) + E_{dir}(v_i),$$

где $E_{std}(v_i)$ — стандартные составляющие внешней энергии, $E_{dir}(v_i)$ — значение предложенной энергии в точке v_i . Для каждой точки (j, k) в матрице энергии для i -й точки контура значение энергии E_{dir} имеет вид:

$$e_{jk}(v_i) = \frac{\min(|d_{jk}(v_i) - d(v_i)|, |d_{jk}(v_i) - d(v_i) + 2\pi|)}{\pi}, \quad (16)$$

где $d_{jk}(v_i)$ — направление градиента изображения в точке, соответствующей точке (j, k) в матрице жадного алгоритма, $d(v_i)$ — направление градиента в точке контура с номером i . Делением на π это значение масштабируется до диапазона $[0, 1]$.

4.1.7 Количественная оценка качества выделения контуров

Предположим, что мы получили некоторый набор контуров для заданного изображения. Для каждой области Ω , ограниченной контуром, соберем следующую информацию:

$$w(\Omega) = \sum_{p \in \Omega} I^{bin}(p), \quad (17)$$

$$\Omega^{med} = \frac{1}{|\Omega|} \sum_{p \in \Omega} I(p). \quad (18)$$

Здесь $w(\Omega)$ — количество белых точек на бинарной версии исходного изображения I^{bin} внутри области Ω , Ω^{med} — средний цвет в I внутри области Ω .

С помощью $w(A)$ будем обозначать количество белых точек в некотором бинарном изображении A :

$$w(A) = \sum_{x=1}^M \sum_{y=1}^N A(x, y).$$

В качестве областей интереса будем рассматривать те, в которых Ω^{med} выше порога, определенного методом Отсу для исходного изображения, в которых $w(\Omega)$ достаточно близко к размеру области. Обозначим совокупность таких

областей $\overline{\Omega}$. В качестве меры для оценки качества выделения контуров на изображении будем использовать величину η :

$$\eta = \chi \frac{w(\overline{\Omega})}{w(I^{bin})} \left(1 - \frac{|\overline{\Omega}| - w(\overline{\Omega})}{NM - w(I^{bin})} \right), \quad (19)$$

$$\chi = 1 - \frac{w(\overline{G} * I^{bin})}{w(I^{bin})}, \quad (20)$$

где $|\overline{\Omega}|$ — количество точек в области $\overline{\Omega}$. Вторым сомножителем в (19) от выделенных областей не зависит. Он отвечает за то, чтобы в процессе бинаризации G_2 было выделено как можно меньше пикселей, не относящихся непосредственно к границе объектов. Величина χ позволяет отбросить те варианты выбранного порога бинаризации, при котором получается переизбыток выделения границ — т.е. в качестве границы выделено много пикселей, которые к ней на самом деле не относятся. Эта ситуация имеет место в случае, если выбран слишком маленький порог бинаризации.

4.1.8 Поиск оптимального порога бинаризации

В предыдущих разделах были подробно описаны основные этапы выделения контура на изображении для предложенного метода. Однако, описание этапов выполнялось для некоторого порога бинаризации G_2 (результат подавление не максимумов поэлементного перемножения градиента и GVF), который считался известным. В данном разделе, опираясь на предложенный в диссертации способ оценки качества выделения контуров, будет предложен алгоритм поиска этого порога и получения финальной конфигурации контуров.

Предполагается, что искомый порог бинаризации для G_2 ниже порога $\overline{T}$, описанного в разделе 4.1.3. В этом случае, можно обозначить его как $\alpha\overline{T}$, и тогда для поиска этого порога можно перебрать — с некоторым шагом — значения α в диапазоне $[0, 1]$, и для каждого значения α , участвующих в переборе, выполнить следующие шаги.

1. Выполняется бинаризация изображения G_2 с порогом $\alpha\overline{T}$.
2. На полученном бинарном изображении выполняется процедура устранения разрывов, в результате которой получается изображение $\overline{G}$.

3. На изображении $\overline{G}$, предложенным методом отслеживания, выделяются контуры имеющихся на изображении объектов.
4. Для полученного набора контуров вычисляется численная оценка качества выделения E_α .

После нахождения численных оценок для всех значений α выбирается то значение, оценка качества которого максимальна, т.е.

$$\alpha^* = \operatorname{argmax}_{\alpha} E_\alpha.$$

В качестве искомых контуров выбираются контуры, которые были найдены при пороге бинаризации G_2 , равном $\alpha^* \overline{T}$, и алгоритм считается завершённым.

4.2 Эксперименты

Для тестирования метода было использовано 266 рентгенографических изображений, содержащих в себе 532 объекта интереса, часть из которых была взята из датасета OAI, а часть предоставлена Ростовским государственным медицинским университетом. Содержимое изображений состояло из рентгенограмм коленного сустава в боковой и фронтальной проекции, имеющих различное разрешение и качество.

В результате тестирования предложенного алгоритма, корректно были выделены контуры для **84,96%** объектов, несмотря на различия в форме, размерах и степень травмированности сустава (рис. 73). При оценке корректности происходило сравнение результата работы алгоритма с контурами, выделенными экспертом, и результат признавался успешным, если наблюдались лишь незначительные отклонения от эталона, не затрагивающие серьёзные изменение формы объектов, разбиение объекта на несколько частей и т. п.

Для **5,08%** объектов результат работы метода был признан удовлетворительным (рис. 74, а–б). В данных случаях, во многом из-за плохого качества рентгенограмм, либо были выделены корректные контуры лишь для части изображений, либо несколько объектов были приняты за один. Предполагается, что подобные результаты в дальнейшем могут быть улучшены при помощи других методов — специально настроенных методов активных контуров и т.п. Для

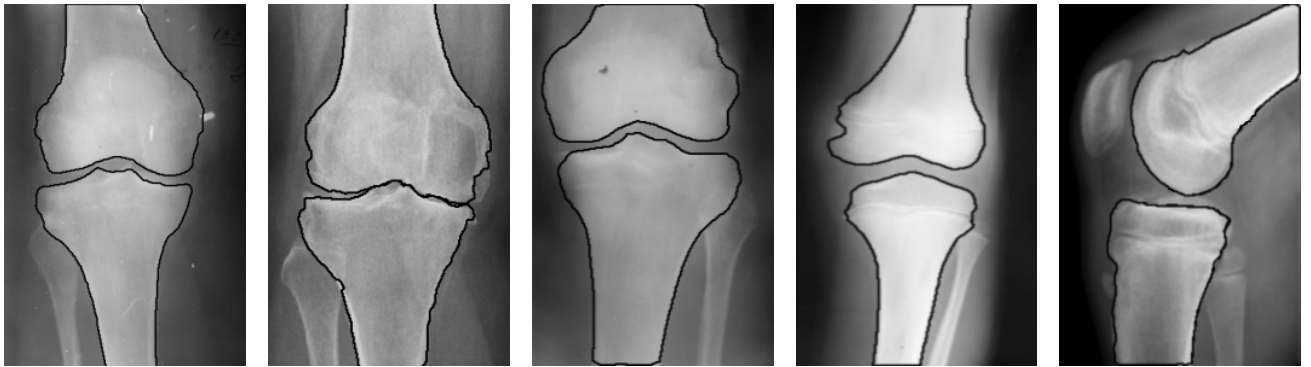


Рисунок 73 — Примеры рентгенограмм коленного сустава в фронтальной и боковой проекциях, для которых предложенный алгоритм сумел корректно выделить искомые контуры

9,96% изображений предложенный алгоритм не смог показать приемлемый результат детектирования (рис. 74, в–г). Основной причиной неудачи при этом является как плохое качество рентгенограмм, так и особенности расположения рентгенографических проекций объектов на изображении, а основной ошибкой детектирования — объединение контуров большеберцовой и бедренной костей в один контур (рис. 74, г).

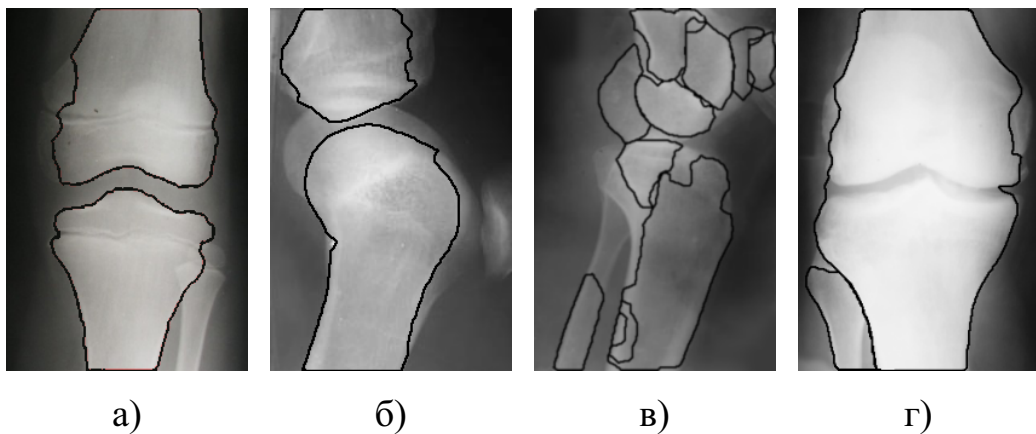


Рисунок 74 — Примеры некорректных результатов работы метода: детектирование ложных границ (а, б), пересегментация (в) и слияние контуров костей (г)

На рис. 75 продемонстрирована работа метода на изображениях, отличающихся от стандартных рентгенограмм коленного сустава. К таким изображениям относятся изображения с горизонтально расположенными суставами вопреки стандартной вертикальной ориентации, примеры детских суставов, вывихов, переломов, а также рентгенограммы предплечий и кистей рук. Показанные примеры показывают универсальность метода и возможность его применения для выделения контуров самых разнообразных объектов, в т.ч. объектов нестандартной формы или объектов с непредсказуемыми изменениями формы.

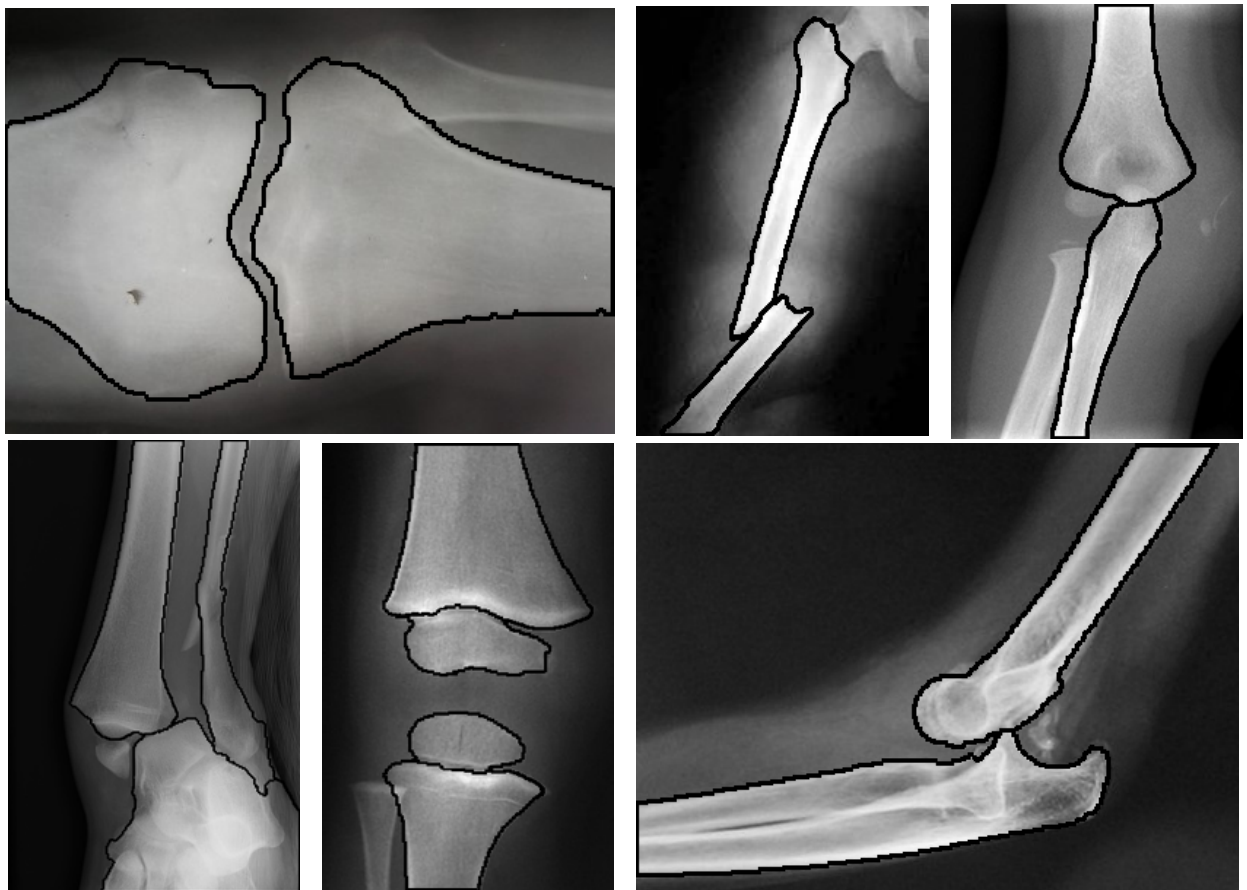


Рисунок 75 — Пример работы предложенного метода для нестандартных изображений костей

Время работы метода для изображений размером около 300×500 , при обработке на центральном процессоре Intel Core i5, не превышало 1–2 секунд на одно изображение. Кроме того, этап поиска оптимального порога бинаризации, описанный в разделе 4.1.8, был распараллелен, что позволило в ряде случаев существенно ускорить время обработки одного изображения.

4.3 Выводы

В соответствии с поставленными задачами в четвертой главе были получены следующие результаты.

1. Предложен новый метод детектирования контуров костей на рентгенограммах, основанный на применении разработанного алгоритма устранения разрывов между фрагментами этих контуров. Метод не требует критерия однородности, что делает его более универсальным,

- и эффективно распараллеливается, что позволяет повысить его производительность. Кроме того, в отличие от методов, основанных на использовании шаблона объекта, предложенный подход применим для детектирования объектов с серьезными нарушениями формы, таких как вывихи, переломы и прочие виды деформаций кости или всего сустава.
2. Разработан алгоритм отслеживания контура замкнутой области, учитывающий градиент изображения.
 3. Предложена новая энергия для метода активных контуров.
 4. Предложена численная оценка качества выделения контуров на изображении, которая используется для автоматического поиска оптимального порога бинаризации на одном из этапов разработанного метода выделения контуров.
 5. Осуществлена программная реализация и тестирование разработанного метода и вспомогательных алгоритмов.

Результаты показывают, что предложенные методы и алгоритмы могут быть использованы для **автоматизированной** диагностики остеоартрита путем автоматизации необходимых действий специалиста или упрощения определенных процедур при добавлении интерактивного режима работы.

В частности, на базе набора контуров может выполняться измерение сужения межсуставного пространства как в ключевых точках (внутренняя и внешняя части кости, центр), так и в виде усредненного значения по всей кромке сустава. Эти значения являются одними из основных численных параметров оценки остеоартрита при проведении исследований. Кроме того, используя контуры, можно измерить степень параллельности поверхности смежных костей в суставе, искажение которой также может свидетельствовать о прогрессировании ОА. В случае ошибок детектирования, специалист может вручную исправить положение ключевых точек контура, и все необходимые параметры будут рассчитаны заново по новой конфигурации точек.

Результаты данной главы опубликованы в [145; 140; 146; 143].

Заключение

Основной научный результат диссертационной работы заключается в решении **актуальной научной задачи**: *разработке методов и алгоритмов повышения точности автоматизированной диагностики остеоартрита по цифровым рентгенограммам без использования априорной информации о форме объектов в условиях ограниченных вычислительных ресурсов.*

При проведении исследований по теме представленной работы получены следующие результаты, обладающие научной новизной:

1. метод локализации суставов на рентгенограмме на базе сверточной нейронной сети специальной архитектуры, **отличающийся** от существующих методов локализации более высокой точностью работы (до 11 раз) и более низкими требованиями к вычислительным ресурсам (вычислительная сложность меньше аналогичных методов в 5-7 раз);
2. метод повышения точности автоматической диагностики остеоартрита нейросетями, использующий комплекс архитектурных решений, **отличающийся** от существующих методов более высокой точностью (повышение точности до 14,39% для модификации шкалы оценки при сохранении информативности диагностики, до 3,85% для интеграции блоков сжатия и возбуждения и до 2,41% при использовании ансамблей) и стабильностью работы;
3. **новый** алгоритм выделения контуров костей на медицинских рентгенографических изображениях, **отличающийся** от существующих возможностью применения для случаев с сильным искажением формы;
4. **новый** алгоритм отслеживания контуров на бинарном изображении, **отличающийся** от существующих большей устойчивостью к артефактам, а также **новая** энергия для метода активных контуров, учитывающая направление градиента при изменении положения точек, что позволяет более точно проводить уточнение контура.

Результаты диссертации были внедрены и использовались в ООО «Наука» (г. Ростов-на-Дону), ООО «Изоскан» (г. Ростов-на-Дону) и в учебном процессе кафедры прикладной математики и программирования (ПМП) Института математики, механики и компьютерных наук (ИММиКН) ЮФУ (г. Ростов-на-Дону).

Предложенный в диссертации комплекс архитектурных решений для повышения точности автоматической диагностики остеоартрита может быть адаптирован и использован для решения задачи повышения точности классификации сверточными нейронными сетями в различных областях. Разработанные методы и алгоритмы могут быть использованы для построения систем интерактивной диагностики и автоматизации действий оператора при проведении измерений на рентгенограммах.

Разработанный программный комплекс может быть использован в медицинских учреждениях, осуществляющих работу с большим потоком данных или учреждениях с недостатком высококвалифицированных кадров, для автоматизации и упрощения процесса диагностики, а также для проведения вручную всех необходимых измерений (измерения углов, расстояний и т.п.).

Все материалы исследования, включая исходный код программного обеспечения и код, использовавшийся для описания и обучения всех упомянутых в работе моделей, доступен публично по ссылке <https://github.com/almikh/automatic-knee-oa-grading-tools>.

Все научные результаты диссертации получены автором лично.

Список литературы

1. Radiographic Assessment of Knee Osteoarthritis / T. Georgiev [и др.] // *Revmatologiya*. — 2016. — Сент. — Т. 24.
2. *Vignon, E. P.* Radiographic issues in imaging the progression of hip and knee osteoarthritis. / E. P. Vignon // *The Journal of rheumatology. Supplement*. — 2004. — Т. 70. — С. 36–44.
3. *Доэрти, М.* Клиническая диагностика болезней суставов / М. Доэрти, Д. Доэрти. — Минск: Тивали, 1993. — 144 с.
4. *Altman, R.* Atlas of individual radiographic features in osteoarthritis, revised. / R. Altman, G. E. Gold // *Osteoarthritis and Cartilage*. — 2007. — Vol. 15 (Suppl A). — A1–56.
5. Reliability and Accuracy of Cross-sectional Radiographic Assessment of Severe Knee Osteoarthritis: Role of Training and Experience / К. М. Klara [и др.] // *The Journal of Rheumatology*. — 2016. — Т. 43. — С. 1421–1426.
6. A radiographic atlas of knee osteoarthritis / J. T. Spector [et al.] // London: Springer Verlag. — 1992. — P. 1–15.
7. Radiographic assessment of progression in osteoarthritis. / R. Altman [et al.] // *Arthritis and Rheumatism*. — 1987. — Vol. 30, no. 11. — P. 1214–1225.
8. *Hart, D.* The classification and assessment of osteoarthritis. / D. Hart, T. D. Spector // *Bailliere's clinical rheumatology*. — 1995. — Vol. 9, no. 2. — P. 407–432.
9. Development of criteria for the classification and reporting of osteoarthritis. Classification of osteoarthritis of the knee / R. Altman [et al.] // *Arthritis and Rheumatism*. — 1986. — Vol. 29, no. 8. — P. 1039–1049. — DOI: [doi:10.1002/art.1780290816](https://doi.org/10.1002/art.1780290816).
10. Defining radiographic knee osteoarthritis: a comparison between the Kellgren & Lawrence classification and oarsi atlas / A. G. Culvenor [et al.] // *Osteoarthritis and Cartilage*. — 2014. — Vol. 22. — P. 256.
11. Microsoft COCO: Common Objects in Context / T.-Y. Lin [et al.] // *Computer Vision – ECCV 2014*. — 2014. — P. 740–755.

12. The Multicenter Osteoarthritis Study: opportunities for rehabilitation research / N. A. Segal [и др.] // PM and R : the journal of injury, function, and rehabilitation. — 2013. — Т. 5, № 8. — С. 647—654. — DOI: [10.1016/j.pmrj.2013.04.014](https://doi.org/10.1016/j.pmrj.2013.04.014).
13. Cohort Profile: Cohort Hip and Cohort Knee (CHECK) study / J. Wesseling [et al.] // International Journal of Epidemiology. — 2016. — Vol. 45, no. 1. — P. 36—44.
14. ImageNet Large Scale Visual Recognition Challenge / O. Russakovsky [и др.] // International Journal of Computer Vision. — 2015. — Т. 115. — С. 211—252.
15. Gradient-based learning applied to document recognition / Y. L. Lecun [и др.] // Proceedings of the IEEE. — 1998. — Т. 86, № 11. — С. 2278—2324. — DOI: [10.1109/5.726791](https://doi.org/10.1109/5.726791).
16. *Krizhevsky, A.* ImageNet Classification with Deep Convolutional Neural Networks / A. Krizhevsky, I. Sutskever, G. Hinton // Advances in neural information processing systems. — 2012. — Jan. — Vol. 25, no. 2. — P. 1097—1105. — DOI: [10.1145/3065386](https://doi.org/10.1145/3065386).
17. *Sánchez, J.* High-dimensional signature compression for large-scale image classification / J. Sánchez, F. Perronnin // Computer Vision and Pattern Recognition (CVPR) 2011. — 2011. — С. 1665—1672.
18. *Nair, V.* Rectified Linear Units Improve Restricted Boltzmann Machines / V. Nair, G. E. Hinton // The 27th International Conference on Machine Learning (ICML-10). — 2010.
19. Improving neural networks by preventing co-adaptation of feature detectors / G. Hinton [и др.] // arXiv. — 2012. — Т. 1207.0580.
20. A survey of the recent architectures of deep convolutional neural networks / A. Khan [и др.] // Artificial Intelligence Review. — 2020. — С. 1—62.
21. *Zeiler, M.* Visualizing and Understanding Convolutional Neural Networks / M. Zeiler, R. Fergus // ECCV 2014, Part I, LNCS 8689. — 2014. — Vol. 8689. — P. 818—833. — DOI: [10.1007/978-3-319-10590-1_53](https://doi.org/10.1007/978-3-319-10590-1_53).
22. *Simonyan, K.* Very Deep Convolutional Networks for Large-Scale Image Recognition / K. Simonyan, A. Zisserman // ArXiv. — 2014. — Vol. abs/1409.1556.

23. *Glorot, X.* Understanding the difficulty of training deep feedforward neural networks / X. Glorot, Y. Bengio // Journal of Machine Learning Research - Proceedings Track. — 2010. — Jan. — Vol. 9. — P. 249–256.
24. *Ioffe, S.* Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift / S. Ioffe, C. Szegedy // ArXiv. — 2015. — Vol. abs/1502.03167.
25. *He, K.* Convolutional neural networks at constrained time cost / K. He, J. Sun // 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2015. — P. 5353–5360. — DOI: [10.1109/CVPR.2015.7299173](https://doi.org/10.1109/CVPR.2015.7299173).
26. Going deeper with convolutions / C. Szegedy [et al.] // 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2015. — P. 1–9.
27. Rethinking the Inception Architecture for Computer Vision / C. Szegedy [et al.] // 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2016. — P. 2818–2826. — DOI: [10.1109/CVPR.2016.308](https://doi.org/10.1109/CVPR.2016.308).
28. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning / C. Szegedy [и др.] // AAAI Conference on Artificial Intelligence. — 2016. — Февр.
29. *Chollet, F.* Xception: Deep Learning with Depthwise Separable Convolutions / F. Chollet // 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2017. — C. 1800–1807.
30. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications / A. G. Howard [et al.] // ArXiv. — 2017. — Vol. abs/1704.04861.
31. Aggregated Residual Transformations for Deep Neural Networks / S. Xie [и др.] // 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2017. — C. 5987–5995.
32. Deep Residual Learning for Image Recognition / K. He [et al.] // 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2016. — P. 770–778. — DOI: [10.1109/CVPR.2016.90](https://doi.org/10.1109/CVPR.2016.90).
33. Identity Mappings in Deep Residual Networks / K. He [и др.] // ArXiv. — 2016. — T. abs/1603.05027.

34. *Huang, G.* Densely Connected Convolutional Networks / G. Huang, Z. Liu, K. Q. Weinberger // 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2017. — P. 2261–2269.
35. *Hu, J.* Squeeze-and-Excitation Networks / J. Hu, L. Shen, G. Sun // 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. — 2018. — C. 7132–7141. — DOI: [doi:10.1109/CVPR.2018.00745](https://doi.org/10.1109/CVPR.2018.00745).
36. Construction of Apple Leaf Diseases Identification Networks Based on Xception Fused by SE Module / X. Chao [и др.] // Applied Sciences. — 2021. — T. 11. — C. 4614.
37. *Sokolova, M.* Beyond Accuracy, F-Score and ROC: A Family of Discriminant Measures for Performance Evaluation / M. Sokolova, N. Japkowicz, S. Szpakowicz // Advances in Artificial Intelligence. AI 2006. Lecture Notes in Computer Science. — 2006. — T. 4304. — DOI: [10.1007/11941439_114](https://doi.org/10.1007/11941439_114).
38. *Tharwat, A.* Classification assessment methods / A. Tharwat // Applied Computing and Informatics. — 2021. — T. 17, № 1. — C. 168–192.
39. *Rezvantlab, A.* Dermatologist Level Dermoscopy Skin Cancer Classification Using Different Deep Learning Convolutional Neural Networks Algorithms / A. Rezvantlab, H. Safigholi, S. Karimijeshni // ArXiv. — 2018. — T. abs/1810.10348.
40. *Fleiss, J. L.* Large sample standard errors of kappa and weighted kappa. / J. L. Fleiss, J. Cohen, B. S. Everitt // Psychological Bulletin. — 1969. — T. 72. — C. 323–327.
41. *Fawcett, T.* ROC Graphs: Notes and Practical Considerations for Researchers / T. Fawcett // Machine Learning. — 2004. — Jan. — Vol. 31. — P. 1–38.
42. *Spackman, K. A.* Signal Detection Theory: Valuable Tools for Evaluating Inductive Learning / K. A. Spackman // ML Workshop. — 1989. — P. 160–163.
43. *Gneiting, T.* Receiver Operating Characteristic (ROC) Curves / T. Gneiting, P. Vogel // arXiv: Methodology. — 2018. — Vol. abs/1809.04808.
44. *Tharwat, A.* Classification of Toxicity Effects of Biotransformed Hepatic Drugs Using Optimized Support Vector Machine / A. Tharwat, T. Gabel, A. E. Hassanien // Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2017. — 2018. — C. 161–170. — DOI: [10.1007/978-3-319-64861-3_15](https://doi.org/10.1007/978-3-319-64861-3_15).

45. *Donoghue, C.* Analysis of MRI for Knee Osteoarthritis using Machine Learning : phdautoref / Donoghue C. — 2013.
46. Deep Learning for Generic Object Detection: A Survey / L. Liu [и др.] // International Journal of Computer Vision. — 2019. — Т. 128. — С. 261—318.
47. Automatic Segmentation of Impaired Joint Space Area for Osteoarthritis Knee on X-ray Image using Gabor Filter Based Morphology Process / L. Anifah [et al.] // IPTEK: The Journal for Technology and Science. — 2011. — Vol. 22, no. 3.
48. Osteoarthritis Classification Using Self Organizing Map Based on Gabor Kernel and Contrast-Limited Adaptive Histogram Equalization / L. Anifah [et al.] // The Open Biomedical Engineering Journal. — 2013. — Vol. 7. — P. 18—28. — DOI: [10.2174/1874120701307010018](https://doi.org/10.2174/1874120701307010018).
49. *González, R. C.* Digital image processing, 3rd Edition / R. C. González, R. E. Woods. — 2008.
50. *Hegadi, R.* Identification of Knee Osteoarthritis Using Texture Analysis / R. Hegadi, U. P. Chavan, D. I. Navale // Data Analytics and Learning. Lecture Notes in Networks and Systems. — 2019. — Vol. 43. — P. 121—129.
51. *Frondelius, T.* Automatic Knee Joint Space Measurement from Plain Radiographs / T. Frondelius. — 2020.
52. *Canny, J.* A Computational Approach to Edge Detection / J. Canny // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 1986. — Т. PAMI—8. — С. 679—698.
53. A Novel Method for Automatic Localization of Joint Area on Knee Plain Radiographs / A. Tiulpin [et al.] // ArXiv. — 2017. — Vol. abs/1701.08991.
54. *Viola, P.* Rapid object detection using a boosted cascade of simple features / P. Viola, M. Jones // Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001. — 2001. — Т. 1. — С. I—I. — DOI: [10.1109/CVPR.2001.990517](https://doi.org/10.1109/CVPR.2001.990517).
55. Automatic knee Kellgren Lawrence grading with artificial intelligence / B. Norman [et al.] // Osteoarthritis and Cartilage. — 2018. — Vol. 26.

56. Knee X-Ray Image Analysis Method for Automated Detection of Osteoarthritis / L. Shamir [et al.] // IEEE Transactions on Biomedical Engineering. — 2009. — Vol. 56, no. 2. — P. 407–415.
57. *Dalal, N.* Histograms of Oriented Gradients for Human Detection / N. Dalal, B. Triggs // In CVPR. — 2005. — C. 886–893.
58. *Lowe, D. G.* Object recognition from local scale-invariant features / D. G. Lowe // Proceedings of the Seventh IEEE International Conference on Computer Vision. — 1999. — T. 2. — C. 1150–1157.
59. Automatic detection of anatomical landmarks on the knee joint using MRI data / N. Xue [и др.] // Journal of Magnetic Resonance Imaging. — 2015. — T. 41.
60. Quantifying radiographic knee osteoarthritis severity using deep convolutional neural networks / J. Antony [et al.] // 2016 23rd International Conference on Pattern Recognition (ICPR). — 2016. — P. 1195–1200.
61. *Bhat, A. Y.* Automated Detection For The Severity Of Knee Osteoarthritis From Plain Radiographs Using Machine Learning Methods / A. Y. Bhat, A.Suhasini // International Journal of Scientific and Technology Research. — 2019. — Vol. 8. — P. 1112–1124.
62. Automated detection of unimpaired joint space for knee osteoarthritis assessment / T. L. Mengko [et al.] // Proceedings of 7th International Workshop on Enterprise networking and Computing in Healthcare Industry, 2005. HEALTHCOM 2005. — 2005. — P. 400–403.
63. OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks / P. Sermanet [и др.] // ArXiv. — 2014. — T. abs/1312.6229.
64. Selective Search for Object Recognition / K. E. A. Sande [и др.] // International Journal of Computer Vision. — 2013. — T. 104. — C. 154–171.
65. Automatic Detection of Knee Joints and Quantification of Knee Osteoarthritis Severity Using Convolutional Neural Networks / J. Antony [et al.] // Machine Learning and Data Mining in Pattern Recognition. MLDM 2017. Lecture Notes in Computer Science. — 2017. — July. — Vol. 10358. — DOI: [10.1007/978-3-319-62416-7_27](https://doi.org/10.1007/978-3-319-62416-7_27).

66. *Long, J.* Fully convolutional networks for semantic segmentation / J. Long, E. Shelhamer, T. Darrell // 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2015. — P. 3431–3440.
67. An Automatic Knee Osteoarthritis Diagnosis Method Based on Deep Learning: Data from the Osteoarthritis Initiative / A. Dogra [и др.] // Journal of Healthcare Engineering. — 2021. — Т. 2021. — DOI: [10.1155/2021/5586529](https://doi.org/10.1155/2021/5586529).
68. Fully automatic knee osteoarthritis severity grading using deep neural networks with a novel ordinal loss / P. Chen [et al.] // Computerized Medical Imaging and Graphics: the official journal of the Computerized Medical Imaging Society. — 2019. — Vol. 75. — P. 84–92. — DOI: [10.1016/j.compmedimag.2019.06.002](https://doi.org/10.1016/j.compmedimag.2019.06.002).
69. Machine Learning in Knee Osteoarthritis: A Review / K. Christos [и др.] // Osteoarthritis and Cartilage Open. — 2020. — Т. 2. — DOI: [10.1016/j.ocarto.2020.100069](https://doi.org/10.1016/j.ocarto.2020.100069).
70. Knee Osteoarthritis Diagnosis Using Support Vector Machine and Probabilistic Neural Network / Ö. Y. Aksehirli [et al.] // International Journal of Computer Science Issues. — 2013. — Vol. 10, no. 1.
71. A novel method for determining the Femoral-Tibial Angle of Knee Osteoarthritis on X-ray radiographs: data from the Osteoarthritis Initiative / R. Wahyuningrum [и др.] // Heliyon. — 2020. — Т. 6, № 8. — С. 471–477. — DOI: [10.1016/j.heliyon.2020.e04433](https://doi.org/10.1016/j.heliyon.2020.e04433).
72. *Kawathekar, P. P.* Severity analysis of Osteoarthritis of knee joint from X-ray images: A Literature review / P. P. Kawathekar, K. J. Karande // 2014 International Conference on Signal Propagation and Computer Technology (ICSPCT 2014). — 2014. — P. 648–652.
73. Fully automatic quantification of knee osteoarthritis severity on plain radiographs. / H. Oka [et al.] // Osteoarthritis and Cartilage. — 2008. — Vol. 16, no. 11. — P. 1300–1306.
74. Normal and threshold values of radiographic parameters for knee osteoarthritis using a computer-assisted measuring system (KOACAD): the ROAD study / H. Oka [et al.] // Journal of Orthopaedic Science. — 2010. — Vol. 15, no. 6. — P. 781–789.

75. Semi-automated digital image analysis of joint space width in knee radiographs. / J. E. Schmidt [et al.] // *Skeletal Radiology*. — 2005. — Vol. 34, no. 10. — P. 639–643.
76. *Pandey, M. S.* Detection of Knee Osteoarthritis Using X-Ray / M. S. Pandey, S. Singh Soam, S. P. Tripathi // *ACEIT Conference Proceeding*. — 2016.
77. *Mikhaylichenko, A.* Automatic Detection of Bone Contours in X-Ray Images / A. Mikhaylichenko, Y. Demyanenko, E. Grushko // *5th International Conference on Image Analysis, Social Networks and Texts (AIST)*. — 2016.
78. *Chan, S.* Osteoarthritis Stages Classification to Human Joint Imagery Using Texture Analysis: A Comparative Study on Ten Texture Descriptors / S. Chan, K. Dittakan // *Recent Trends in Image Processing and Pattern Recognition (RTIP2R)*. — 2019. — P. 209–225.
79. A decision support tool for early detection of knee OsteoArthritis using X-ray imaging and machine learning: Data from the OsteoArthritis Initiative / A. Brahim [et al.] // *Computerized medical imaging and graphics: the official journal of the Computerized Medical Imaging Society*. — 2019. — Vol. 73. — P. 11–18.
80. WND-CHARM: Multi-purpose image classification using compound image transforms / N. Orlov [et al.] // *Pattern Recognition Letters*. — 2008. — Vol. 29, no. 11. — P. 1684–1693.
81. *Aggarwal, C. C.* Data Classification: Algorithms and Applications / C. C. Aggarwal // *CRC Press (Data Mining and Knowledge Discovery Series)*. — 2015. — P. 44.
82. Early Detection of Radiographic Knee Osteoarthritis Using Computer-aided Analysis / L. Shamir [et al.] // *Osteoarthritis and Cartilage*. — 2009. — Vol. 17, no. 10. — P. 1307–1312.
83. *Gornale, S. S.* Detection of Osteoarthritis using Knee X-Ray with Random Forest / S. S. Gornale, P. U. Patravali, R. R. Manza // *International Journal of Computer Applications*. — 2016. — Vol. 145, no. 1.
84. Determination of Osteoarthritis Using Histogram of Oriented Gradients and SVM / S. S. Gornale [et al.] // *Image, Graphics and Signal Processing*. — 2017. — Vol. 12. — P. 41–49.

85. Osteoarthritis severity of the hip by computer-aided grading of radiographic images / I. Boniatis [et al.] // Medical and Biological Engineering and Computing. — 2006. — Vol. 44, no. 9. — P. 793–803. — DOI: [10.1007/s11517-006-0096-3](https://doi.org/10.1007/s11517-006-0096-3).
86. *Navale, D. I.* Block based texture analysis approach for knee osteoarthritis identification using SVM / D. I. Navale, R. Hegadi, N. Mendgudli // 2015 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE). — 2015. — P. 338–341.
87. *Minciullo, L.* Fully automated shape analysis for detection of Osteoarthritis from lateral knee radiographs / L. Minciullo, T. F. Cootes // 2016 23rd International Conference on Pattern Recognition (ICPR). — 2016. — P. 3787–3791.
88. Automated Shape and Texture Analysis for Detection of Osteoarthritis from Radiographs of the Knee / J. Thomson [et al.] // Medical Image Computing and Computer-Assisted Intervention (MICCAI). — 2015. — Vol. 9350. — P. 127–134. — DOI: [10.1007/978-3-319-24571-3_16](https://doi.org/10.1007/978-3-319-24571-3_16).
89. A guide to deep learning in healthcare / A. Esteva [и др.] // Nature Medicine. — 2019. — C. 24–29.
90. The Use of Artificial Intelligence in the Evaluation of Knee Pathology. / E. Garwood [et al.] // Seminars in Musculoskeletal Radiology. — 2020. — Vol. 24, no. 1. — P. 21–29. — DOI: [10.1055/s-0039-3400264](https://doi.org/10.1055/s-0039-3400264).
91. *Pratiwi, D.* An Application of Backpropagation Artificial Neural Network Method for Measuring The Severity of Osteoarthritis / D. Pratiwi, D. D. Santika, B. Pardamean // International Journal of Engineering & Technology IJET-IJENS. — 2011. — Vol. abs/1309.7522, no. 3.
92. Feature Learning to Automatically Assess Radiographic Knee Osteoarthritis Severity / J. Antony [и др.] // arXiv. — 2019. — T. abs/1908.08840.
93. Applying Densely Connected Convolutional Neural Networks for Staging Osteoarthritis Severity from Plain Radiographs / B. Norman [и др.] // Journal of Digital Imaging. — 2019. — T. 32, № 3. — C. 471–477. — DOI: [10.1007/s10278-018-0098-3](https://doi.org/10.1007/s10278-018-0098-3).

94. A New Approach to Classify Knee Osteoarthritis Severity from Radiographic Images based on CNN-LSTM Method / R. T. Wahyuningrum [и др.] // 2019 IEEE 10th International Conference on Awareness Science and Technology (iCAST). — 2019. — С. 1—6. — DOI: [10.1109/ICAWSST.2019.8923284](https://doi.org/10.1109/ICAWSST.2019.8923284).
95. Assessing Knee OA Severity with CNN attention-based end-to-end architectures / M. Górriz [и др.] // ArXiv. — 2019. — Т. abs/1908.08856.
96. *Bin, L.* Toward automatic quantification of knee osteoarthritis severity using improved Faster R-CNN / L. Bin, L. Jianxu, H. Huan // International Journal of Computer Assisted Radiology and Surgery. — 2020. — Т. 15, № 3. — С. 457—466. — DOI: [10.1007/s11548-019-02096-9](https://doi.org/10.1007/s11548-019-02096-9).
97. Automatic Knee Osteoarthritis Diagnosis from Plain Radiographs: A Deep Learning-Based Approach / A. Tiulpin [et al.] // Scientific Reports. — 2018. — Vol. 8.
98. *Tiulpin, A.* KNEEL: Knee Anatomical Landmark Localization Using Hourglass Networks / A. Tiulpin, I. Melekhov, S. Saarakkala // 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW). — 2019. — P. 352—361.
99. *Tiulpin, A.* Automatic Grading of Individual Knee Osteoarthritis Features in Plain Radiographs using Deep Convolutional Neural Networks / A. Tiulpin, S. Saarakkala // ArXiv. — 2019. — Vol. abs/1907.08020.
100. Multimodal Machine Learning-based Knee Osteoarthritis Progression Prediction from Plain Radiographs and Clinical Data / A. Tiulpin [et al.] // Scientific Reports. — 2019. — Vol. 9.
101. *Friedman, J.* Greedy function approximation: A gradient boosting machine. / J. Friedman // Annals of Statistics. — 2001. — Vol. 29. — P. 1189—1232.
102. Automated Grading of Radiographic Knee Osteoarthritis Severity Combined with Joint Space Narrowing / H. Gu [и др.] // ArXiv. — 2022. — Т. abs/2203.08914.
103. Machine Learning-Based Automatic Classification of Knee Osteoarthritis Severity Using Gait Data and Radiographic Images / S. B. Kwon [и др.] // IEEE Access. — 2020. — Т. 8. — С. 120597—120603. — DOI: [10.1109/ACCESS.2020.3006335](https://doi.org/10.1109/ACCESS.2020.3006335).

104. *Chernuhin, N. A.* On an approach to object recognition in X-ray medical images and interactive diagnostics process / N. A. Chernuhin // Ninth International Conference on Computer Science and Information Technologies Revised Selected Papers. — 2013. — C. 1–6.
105. Unsupervised connectivity-based thresholding segmentation of midsagittal brain MR images. / C. Lee [и др.] // Computers in biology and medicine. — 1998. — T. 28, № 2. — C. 309–338.
106. *Adams, R.* Seeded Region Growing / R. Adams, L. Bischof // IEEE Trans. Pattern Anal. Mach. Intell. — 1994. — T. 16, № 6. — C. 641–647.
107. *Roerdink, J.* The Watershed Transform: Definitions, Algorithms and Parallelization Strategies / J. Roerdink, A. Meijster // Fundam. Inform. — 2000. — T. 41. — C. 187–228.
108. Graph extraction from color images / T. Lourens [и др.] // In Proceeding European Symposium on Artificial Neural Networks. — 2001. — C. 329–334.
109. Automatic Extraction of Femur Contours from Hip X-Ray Images / Y. Chen [и др.] // Computer Vision for Biomedical Image Applications. — 2005. — C. 200–209.
110. *Williams, D.* A Fast algorithm for active contours and curvature estimation / D. Williams, M. Shah // CVGIP Image Underst. — 1992. — T. 55, № 1. — C. 14–26.
111. Carpal-Bone Feature Extraction Analysis in Skeletal Age Assessment Based on Deformable Model / L. Pan [и др.] // Journal of Computer Science and Technology. — 2004. — T. 4. — C. 152–156.
112. A fully automatic algorithm for contour detection of bones in hand radiographs using active contours / R. D. García [и др.] // Proceedings 2003 International Conference on Image Processing (Cat. No.03CH37429). — 2003. — T. 3. — C. 421–424.
113. Automatic Extraction of Femur Contours from Hip X-Ray Images / Y. Chen [и др.] // Computer Vision for Biomedical Image Applications. — 2005. — C. 200–209.

114. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks / S. Ren [et al.] // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2015. — Vol. 39. — P. 1137–1149. — DOI: [10.1109/TPAMI.2016.2577031](https://doi.org/10.1109/TPAMI.2016.2577031).
115. You Only Look Once: Unified, Real-Time Object Detection / J. Redmon [и др.] // 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2016. — С. 779–788. — DOI: [10.1109/CVPR.2016.91](https://doi.org/10.1109/CVPR.2016.91).
116. SSD: Single Shot MultiBox Detector / W. Liu [et al.] // Computer Vision – ECCV 2016. — 2016. — Oct. — Vol. 9905. — P. 21–37. — DOI: [10.1007/978-3-319-46448-0_2](https://doi.org/10.1007/978-3-319-46448-0_2).
117. Speed/Accuracy Trade-Offs for Modern Convolutional Object Detectors / J. Huang [et al.] // 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2017. — P. 3296–3297.
118. *Ruder, S.* An overview of gradient descent optimization algorithms / S. Ruder // ArXiv. — 2016. — Vol. abs/1609.04747.
119. *Kingma, D.* Adam: A Method for Stochastic Optimization / D. Kingma, J. Ba // The 3rd International Conference on Learning Representations. — 2015.
120. *Гайер, А.* Аугментация обучающей выборки на лету для обучения нейронных сетей / А. Гайер, А. Шешкус, Ю. Чернышова // Труды Института системного анализа Российской академии наук. — 2018. — Т. 68. — С. 150–157.
121. Dropout: a simple way to prevent neural networks from overfitting / N. Srivastava [et al.] // J. Mach. Learn. Res. — 2014. — Vol. 15. — P. 1929–1958.
122. *Torrey, L.* Transfer learning / L. Torrey, J. Shavlik // Handbook of Research on Machine Learning Applications. — 2009. — Jan. — DOI: [10.4018/978-1-60566-766-9.ch011](https://doi.org/10.4018/978-1-60566-766-9.ch011).
123. *Laan, M.* Super Learner / M. Laan, E. Polley, A. Hubbard // Statistical applications in genetics and molecular biology. — 2007. — Feb. — Vol. 6, no. 1. — Article25. — DOI: [10.2202/1544-6115.1309](https://doi.org/10.2202/1544-6115.1309).
124. *Xieping, G.* The Researching Development of Evolutionary Neural Networks / G. Xieping // Systems engineering and electronics. — 2001. — Т. 23, № 10. — С. 92–97.

125. *Chen, G.* Particle Swarm Optimization Neural Network and Its Application in Soft-Sensing Modeling / G. Chen, J. Yu // Lecture notes in computer science. — 2005. — T. 3611. — C. 610–617.
126. *Hansen, L.* Neural Network Ensembles / L. Hansen, P. Salamon // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 1990. — T. 12, № 10. — C. 993–1001.
127. *Breiman, L.* Bagging predictors / L. Breiman // Machine Learning. — 1996. — Vol. 24. — P. 123–140.
128. *Hansen, L. K.* Neural Network Ensembles / L. K. Hansen, P. Salamon // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 1990. — Nov. — Vol. 12, no. 10. — P. 993–1001. — DOI: [10.1109/34.58871](https://doi.org/10.1109/34.58871).
129. *Li, H.* Research and development of neural network ensembles: a survey / H. Li, X. Wang, S. Ding // Artificial Intelligence Review. — 2016. — Vol. 49. — P. 455–479.
130. *Ju, C.* The Relative Performance of Ensemble Methods with Deep Convolutional Neural Networks for Image Classification / C. Ju, A. Bibaut, M. J. van der Laan // Journal of Applied Statistics. — 2018. — Vol. 45. — P. 2800–2818. — DOI: [10.1080/02664763.2018.1441383](https://doi.org/10.1080/02664763.2018.1441383).
131. *Freund, Y.* A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting / Y. Freund, R. Schapire // Journal of Computer and System Sciences. — 1999. — Dec. — Vol. 55. — P. 119–139. — DOI: [10.1006/jcss.1997.1504](https://doi.org/10.1006/jcss.1997.1504).
132. *Tomasi, C.* Bilateral filtering for gray and color images / C. Tomasi, R. Manduchi // Sixth International Conference on Computer Vision. — 1998. — C. 839–846.
133. *Kornprobst, P.* Bilateral Filtering: Theory and Applications / P. Kornprobst, J. Tumblin, F. Durand // Found. Trends Comput. Graph. Vis. — 2009. — T. 4. — C. 1–74.
134. Fast Bilateral Filtering for the Display of High-Dynamic-Range Images / Y.-Y. Chuang [и др.] // ACM Transactions on Graphics. — 2002. — T. 21, № 3. — C. 257–266.

135. *Otsu, N.* A Threshold Selection Method from Gray-Level Histograms / N. Otsu // IEEE Transactions on Systems, Man, and Cybernetics. — 1979. — Т. 9. — С. 62–66.
136. *Kirsch, R.* Computer determination of the constituent structure of biological images. / R. Kirsch // Computers and biomedical research, an international journal. — 1971. — Т. 4, № 3. — С. 315–28.
137. *Xu, C.* Snakes, shapes, and gradient vector flow / C. Xu, J. Prince // IEEE transactions on image processing: a publication of the IEEE Signal Processing Society. — 1998. — Т. 7, № 3. — С. 359–69.
138. *Петров, В. О.* Модификация алгоритма активных контуров для решения задачи интерактивной сегментации растровых изображений дефектов металлических отливок / В. О. Петров, О. О. Привалов // Современные проблемы науки и образования. — 2008. — Т. 6. — С. 14–19.

Публикации автора по теме научно-квалификационной работы

В изданиях из списка ВАК РФ

139. *Михайличенко, А. А.* Аналитический обзор методов оценки качества алгоритмов классификации в задачах машинного обучения / А. А. Михайличенко // Вестник Адыгейского государственного университета. Серия 4: Естественно-математические и технические науки. — 2022. — Т. 311, № 4. — С. 52–59. — DOI: [10.53598/2410-3225-2022-4-311-52-59](https://doi.org/10.53598/2410-3225-2022-4-311-52-59).

В изданиях, входящих в международную базу цитирования Scopus

140. *Mikhaylichenko, A.* Automatic Detection of Bone Contours in X-Ray Images / A. Mikhaylichenko, Y. Demyanenko, E. Grushko // CEUR Workshop Proceedings. Proceedings on 5th International Conference on Image Analysis, Social Networks and Texts. — 2016. — Vol. 1710. — P. 212–223. — Access mode: <https://ceur-ws.org/Vol-1710/paper21.pdf> (access mode 30.06.2023).

141. *Mikhaylichenko, A.* Automatic Grading of Knee Osteoarthritis from Plain Radiographs Using Densely Connected Convolutional Networks / A. Mikhaylichenko, Y. Demyanenko // Recent Trends in Analysis of Images, Social Networks and Texts. AIST 2020. Communications in Computer and Information Science. — 2021. — Vol. 1357. — P. 149–161. — DOI: [10.1007/978-3-030-71214-3_13](https://doi.org/10.1007/978-3-030-71214-3_13).
142. *Михайличенко, А. А.* Использование блоков сжатия и возбуждения для повышения точности автоматической классификации остеоартрита коленного сустава при помощи сверточных нейронных сетей / А. А. Михайличенко, Я. М. Демяненко // Компьютерная оптика. — 2022. — Т. 46, № 2. — С. 317–325. — DOI: [10.18287/2412-6179-CO-897](https://doi.org/10.18287/2412-6179-CO-897).
143. *Михайличенко, А. А.* Выделение контуров костей коленного сустава на медицинских рентгенограммах / А. А. Михайличенко, Я. М. Демяненко // Компьютерная оптика. — 2019. — Т. 43, № 3. — С. 455–463. — DOI: [10.18287/2412-6179-2019-43-3-455-463](https://doi.org/10.18287/2412-6179-2019-43-3-455-463).

В сборниках трудов конференций

144. *Михайличенко, А. А.* Визуализация мотивов сверточной нейронной сети при принятии решения на примере автоматической классификации стадии остеоартрита / А. А. Михайличенко, Я. М. Демяненко // Современные информационные технологии: тенденции и перспективы развития. Материалы XXVIII научной конференции (Южный федеральный университет, Ростов-на-Дону, 13–15 мая 2021 г.) — Ростов-на-Дону; Таганрог: Издательство Южного федерального университета, 2021. — С. 257–259.
145. *Михайличенко, А. А.* Метод автоматического детектирования объектов на медицинских рентгенографических изображениях / А. А. Михайличенко, Я. М. Демяненко // Осенние математические чтения в Адыгее: материалы I Международной научной конференции, посвященной памяти профессора Казбека Сагидовича Мамия, 8–10 октября 2015 г. — Майкоп: АГУ, 2015. — С. 147–149.

146. *Михайличенко, А. А.* Метод точного выделения контуров медицинских объектов на рентгенограммах / А. А. Михайличенко, Я. М. Демяненко // Графикон 2016 : труды 26-й Международной научной конференции, Россия, Нижний Новгород, 19–23 сентября 2016 г. — Нижний Новгород: Институт физико-технической информатики, 2016. — С. 193–198.
147. *Михайличенко, А. А.* Автоматическая локализация коленного сустава на рентгенограмме для оценки развития остеоартрита / А. А. Михайличенко // Современные информационные технологии: тенденции и перспективы развития. Материалы XXVII научной конференции (Южный федеральный университет, Ростов-на-Дону, 24–26 сентября 2020 г.) — Ростов-на-Дону; Таганрог: Издательство Южного федерального университета, 2020. — С. 189–191.
148. *Михайличенко, А. А.* Методы оценки качества работы бинарных и мультатлассовых классификаторов / А. А. Михайличенко // Современные информационные технологии: тенденции и перспективы развития : материалы XXIX научной конференции (Южный федеральный университет, Ростов-на-Дону, 21–23 апреля 2022 г.). — Ростов-на-Дону; Таганрог: Издательство Южного федерального университета, 2022. — С. 198–203.

Список рисунков

1	Измерение сужения межсуставного пространства в специализированном программном обеспечении	16
2	Система Келлгрена-Лоуренса для оценки прогрессирования остеоартрита коленного сустава	19
3	Примеры полных рентгенограмм коленей в двухсторонней задне-передней проекции с фиксированным сгибанием	21
4	Примеры изображений области коленного сустава, полученных в результате разметки из исходных рентгенограмм	22
5	Примеры данных из датасета CHECK, слева направо: передне-задняя проекция таза, боковая, фронтальная и аксиальная проекции коленного сустава, прямая проекция кистей	24
6	Пример данных, исключенных из датасета CHECK	24
7	LeNet-5	26
8	Ошибка обучения (слева) и ошибка теста (справа) на CIFAR-10 с 20-ти и 56-ти слойными «простыми» сетями; более глубокая сеть имеет большую ошибку обучения и, следовательно, ошибку тестирования	29
9	Примеры пары слоев просто нейронной сети (слева) и блока в сети архитектуры ResNet (справа)	30
10	Пример остаточной функции F из работы [24]: базовый блок для слоя размером 56×56 для ResNet-34 (слева) и расширенный, или bottleneck, блок для ResNet-50 и выше	31
11	«Наивная» версия блока Inception (слева) и блок с понижением размерности (справа) из одноименной архитектуры. Понижение размерности производится свертками 1×1	33
12	Блок Inception, в котором свертка 5×5 заменена на две последовательные свертки 3×3	34
13	Inception-блок после факторизации свертки $N \times N$	34
14	Inception-блок с повышенным количеством сверток на одном уровне вычислений	35
15	Модули Inception-A, Inception-B и Inception-C для Inception-Resnet . .	36

16	Редукционные блоки для архитектуры Inception-ResNet, понижающий вдвое пространственную размерность данных. Разные блоки используются на разных уровнях сети	37
17	Иллюстрация базовой версии архитектуры Inception-Resnet-v1 из работы [28]. Блок <i>Steam</i> обозначает последовательность из нескольких сверток с пуллингом, понижающих начальную размерность $299 \times 299 \times 3$ до $35 \times 35 \times 256$ на выходе из блока. Reduction-A, Reduction-B — различные вариации редукционных блоков	38
18	Упрощенная версия Inception-модуля (a) и эквивалентная ей реорганизованная версия (b)	39
19	Экстремальная версия Inception-модуля	39
20	Внешний вид базовой архитектуры Xception	40
21	Иллюстрация базового блока ResNet (слева) и базового блока ResNeXt с кардинальностью 32 (справа)	42
22	DenseNet с двумя плотно-связанными блоками. Слои между двумя смежными блоками называются переходными слоями и меняют размеры карты признаков посредством свертки и пуллинга. В качестве перехода от сверточных слоев к полносвязному используется GAP	44
23	Иллюстрация блока сжатия и возбуждения	46
24	Интеграция SE-блоков в архитектуру Inception: исходный модуль (слева) и модуль с добавленным блоком сжатия и возбуждения (справа); в качестве коэффициента редукции r использовалось значение 16	48
25	Интеграция SE-блоков в архитектуры ResNet и ResNeXt: исходный модуль (слева) и модуль с блоком сжатия и возбуждения (справа) . . .	49
26	Интеграция SE-блоков в архитектуру Xception: исходный модуль (слева) и модуль с добавленным блоком сжатия и возбуждения (справа)	50
27	Интеграция SE-блоков в архитектуру DenseNet на примере плотно-связанного блока	51
28	Иллюстрация матрицы неточностей для бинарной классификации . . .	53
29	Иллюстрация матрицы неточностей для мультиклассовой классификации, для случая трех классов	58

30	Иллюстрация ROC-кривой с указанием важных для оценки классификатора точек, а также оптимистичный, пессимистичный и ожидаемый сегменты	60
31	Пример вычисления гистограммы по изображению, после применения к нему фильтра Габора [48] (слева) и пример области сустава, найденной при помощи гистограммы по бинарному изображению из работы [50]	62
32	Пример вычисления гистограммы по изображению, после применения к нему фильтра Габора [48] (слева) и пример области сустава, найденной при помощи гистограммы по бинарному изображению из работы [50]	63
33	Пример лучшего центра сустава (синий цвет) и финальной области сустава (зеленый цвет), выбранные после применения классификатора ко всем кандидатам из работы [60]	65
34	Архитектура полностью сверточной нейронной сети для локализации коленного сустава из работы [65]	67
35	Схема вычисления множества признаков из исходного изображения в методе WND-CHARM из оригинальной работы, многие признаки вычисляются повторно над различными преобразованиями исходного изображения — картой границ, вейвлет-преобразованием и др.	71
36	Пример одной из первых архитектур нейронной сети из работы [65], используемой для классификации и регрессии стадии остеоартрита по шкале Келлгрена-Лоуренса	75
37	Пример интеграции модулей внимания в архитектуру VGG-16 из работы [95]	77
38	Пример сиамской нейронной сети из работы [97]. Из имеющейся рентгенограммы выбираются два патча и подаются на вход различным веткам сети. Синие блоки на изображении — стандартная комбинация сверточного слоя, нормализации батчей (BN) и функции активации ReLU, серые круги — слои пуллинга, светло-красные блоки — блоки глобального усредняющего пуллинга, а зеленый блок — слой Softmax	78

39	Общая схема метода из работы [99]. Сеть предсказывает степень ОА по атласу OARSI для бедренной боковой (FL), большеберцовая боковой (TL), бедренной медианной (FM) и большеберцовый медианной частей кости (TM), а также сужение межсуставного пространства для бокового (JSN L) и медианного (JSN M) отделов сустава	79
40	Крупный вид участка бедренного сустава с неоднородной структурой (а) и участок с перекрытием двух костей (б)	81
41	Пример сегментации костей кисти руки из работы [111]: начальный контур (а) и результат применения алгоритма активных контуров (б) .	83
42	Примеры элементов для поиска начальной позиции контуров (а), неточного описания шаблоном формы реального объекта (б) и пример нестандартной формы бедренного сустава, что не позволило его идентифицировать при помощи имеющихся шаблонов (в)	84
43	Иллюстрация выбора положений для исследования наличия объектов на карте признаков. Слева — искомые ограничивающие прямоугольники, в середине — разбиение карты признаков сеткой 8x8, справа — разбиение сеткой 4x4 и параметры искомого прямоугольника	89
44	Иллюстрация нескольких вариантов локализации для каждого сустава методом SSD (до подавления не максимумов)	90
45	Пример порядковой матрицы W из [68], которая также используется и в данной работе	99
46	Визуализация матриц ошибок для классификаторов WND-CHARM с коэффициентом масштабирования 0,5, 0,75 и 1,0 от исходного размера изображения соответственно (слева направо)	103
47	Матрица ошибок (слева) и ROC-кривая (справа), полученные в результате экспериментов для модели из работы [65]. На графике ROC-кривой также указаны значения площади под кривой (AUC) для каждой из степеней ОА	107
48	График обучения сети из работы [65]. По оси X отложен номер эпохи, по оси Y — точность. Оранжевый график — точность работы модели на тренировочной выборке, синий — на валидационной	108

49	Влияние блоков сжатия и возбуждения на качество работы сверточных сетей традиционных архитектур для полной шкалы Келлгрена-Лоуренса	111
50	Визуализация матриц ошибок для SE-ResNeXt-50 (слева) и Inception-ResnetV2 (справа)	112
51	Влияние блоков сжатия и возбуждения на качество работы сверточных сетей традиционных архитектур для сокращенной шкалы Келлгрена-Лоуренса	114
52	Визуализация матриц ошибок для SE-DenseNet-121 (слева) и Inception (справа)	115
53	Матрица ошибок (слева) и ROC-кривая (справа) для наилучшего экземпляра SE-DenseNet-121, обученного для решения задачи бинарной классификации. Класс 0 означает отсутствие ОА, класс 1 — наличие признаков ОА на рентгенограмме	118
54	Диаграмма супер-классификатора из работы [123]	122
55	Матрица ошибок (слева) и ROC-кривая (справа) для ансамбля сетей DenseNet-121	126
56	Матрица ошибок (слева) и ROC-кривая (справа) для ансамбля сетей SE-Xception для задачи бинарной классификации	128
57	Влияние использования порядковой функции потерь при обучении на финальное качество работы сверточных сетей	130
58	Изменение точности классификации ОА коленного сустава при помощи известных архитектур сверточных нейронных сетей при сокращении шкалы Келлгрена-Лоуренса до четырехклассовой (KL-4) и двухклассовой (KL-2)	130
59	Улучшение точности классификации при использовании ансамблей	131
60	Иллюстрация билатерального фильтра из работы [134]. Весовые коэффициенты применяются к пикселю под стрелкой	137
61	Пример бинаризации изображения с порогом, вычисленным методом Отсу, для получения примерной маски объектов	138

- 62 Иллюстрация к этапу вычисления градиента: значение модуля градиента сглаженной версии исходного изображения $|\nabla I|$, вычисленное при помощи оператора Кирша (а) для участка изображения и векторное поле потока градиента для того же участка (б) 139
- 63 Иллюстрация этапа формирования «сильных» фрагментов границ: результат поэлементного перемножения модулей градиента и GVF (а), результат применения подавления не максимумов к изображению G_1 (б) и результат бинаризации G_2 с некоторым порогом из множества $\hat{T}$ (в) 140
- 64 Примеры точек разрыва: точки разрыва первого (а), второго (б) и третьего (в) типов 142
- 65 Пример поэтапного устранения разрывов границ: слева направо показано состояние изображения для 0, 25 и 55 итерации предложенного алгоритма соответственно 143
- 66 Иллюстрация к условию существования разрыва между точками p и q . В данном случае условие выполняется — q лежит между $\bar{v}$ и $\bar{w}$. . 144
- 67 Нарушение условия о не пересечении существующей границы (слева) и пример устранения разрыва (справа) 145
- 68 Пример устранения разрыва для условия 15 (слева) и устранение разрыва для условия 14 (справа) 146
- 69 Общая идея алгоритма отслеживания контура 148
- 70 Иллюстрация преимущества предлагаемого метода на примере участка с возможностью неоднозначной трактовки границы: пример проблемного участка бинарного изображения (а), фрагмент контура, найденный одним из стандартных алгоритмов отслеживания контуров — «алгоритмом жука» (б) и фрагмент контура, найденный предложенным методом с использованием карты градиента (в) 149
- 71 Иллюстрация жадного алгоритма в методе активных контуров из работы [138]: в качестве текущей точки рассматривается v_i, v'_i — изменение положения для минимизации общей энергии всего контура 151
- 72 Иллюстрация действия новой энергии E_{dir} . Стрелками схематично обозначено направление градиента на представленном фрагменте изображения — в данном случае $E_{dir}(v'_i) > E_{dir}(v''_i)$ 151

73	Примеры рентгенограмм коленного сустава в фронтальной и боковой проекциях, для которых предложенный алгоритм сумел корректно выделить искомые контуры	155
74	Примеры некорректных результатов работы метода: детектирование ложных границ (а, б), пересегментация (в) и слияние контуров костей (г)	155
75	Пример работы предложенного метода для нестандартных изображений костей	156
76	Иллюстрация панели инструментов	215
77	Иллюстрация угла Кобба	216
78	Иллюстрация измерительных инструментов, доступных в приложении: измерение длины (1), измерение угла (2), эллиптическая область (3), угол Кобба (4), многоугольная область (5) и умная кривая (6)	217
79	Иллюстрация основного интерфейса программного обеспечения: вид после автоматической классификации остеоартрита	222

Список таблиц

1	Классификация остеоартрита по Косинской	17
2	Стадии развития ОА по шкале Келлгрена-Лоуренса	18
3	Распределение изображений датасета ОАI по возрасту и полу	21
4	Распределение изображений, использовавшихся для экспериментов. Числа в таблице обозначают число изображений суставов, имеющихся в каждой из групп	23
5	Архитектура сети MobileNet: Conv - обычные свертки, conv dw - свертки, разделимые по глубине (depthwise separable convolutions) . . .	93
6	Сравнение методов автоматического детектирования, основанное на коэффициенте Жаккарда (J)	94
7	Распределение изображений, использовавшихся для эксперимента с WND-CHARM, по стадиям остеоартрита шкалы Келлгрена-Лоуренса .	102
8	Точность классификации методом WND-CHARM для различных вариантов понижения оригинальной размерности (1.0 — оригинальный размер изображений)	103
9	Метрики качества работы метода WND-CHARM для всех использовавшихся в эксперименте коэффициентов понижения размерности	104
10	Архитектура сети из работы [65]. Размер входного изображения — 300 × 200, аббревиатура conv означает сверточный слой, pool — слой пуллинга, fc — полносвязный слой	105
11	Метрики результатов тренировки сети как для модели из проведенного эксперимента, так и для аналогичной модели из оригинальной работы	107
12	Среднеклассовые метрики качества работы различных классификаторов для разных функций потерь (усредненные по трем экспериментам)	110
13	Подробные метрики качества работы классификаторов, показавших наилучшие результаты работы для каждого из классов (усредненные по трем экспериментам)	112

14	Среднеклассовые метрики качества работы различных классификаторов для разных функций потерь (усредненные по трем экспериментам)	113
15	Подробные метрики качества работы классификаторов, показавших наилучшие результаты работы, для каждого из классов (усредненные по трем экспериментам)	115
16	Количество изображений в тренировочной, валидационной и тестовой выборках. Non-OA — рентгенограммы здорового коленного сустава, OA — рентгенограммы, имеющие признаки остеоартрита . .	116
17	Среднеклассовые метрики качества работы различных классификаторов для разных функций потерь (усредненные по трем экспериментам)	117
18	Среднеклассовые метрики качества работы различных классификаторов для разных функций потерь (усредненные по трем экспериментам)	124
19	Среднеклассовые метрики качества работы различных классификаторов (4 класса) для разных функций потерь (усредненные по трем экспериментам)	125
20	Результаты работы различных архитектур на задаче определения факта наличия остеоартрита на рентгенограмме — по каждому из пунктов показаны усредненные данные для нескольких моделей, обученных с разным случайным начальным значением	127
21	Метрики результатов ансамбля сетей SE-Xception для задачи бинарной классификации	128
22	Изменение количества параметров модели при добавлении SE-блоков в известные архитектуры	129
23	Сравнение методов автоматической диагностики остеоартрита для разных методов оценки	132
24	Значение коэффициента k для лучших моделей, полученных в рамках диссертации, по всем используемым шкалам оценки OA из таблицы 23	134
25	Порядок применения алгоритмов, тестовые значения ограничений и коэффициентов	147

26	Структура сверточных нейронных сетей с архитектурой ResNet различной глубины	188
27	Структура сети с архитектурой ResNeXt, C — кардинальность соответствующего блока	189
28	Структура сети с архитектурой DenseNet-121. Каждый сверточный слой плотно-связанного (Dense Block) или переходного (Transition) при этом сопровождается использованием нормализации батчей и нелинейностью в виде ReLU	190
29	Структура сетей ResNet и ResNeXt различной глубины с интегрированными SE-блоками	191
30	Структура сети с архитектурой SE-DenseNet-121	192
31	Подробные метрики для DenseNet-121, 5 классов	193
32	Подробные метрики для Inception, 5 классов	193
33	Подробные метрики для InceptionResNetV2, 5 классов	194
34	Подробные метрики для InceptionResNetV2, 2 класса	194
35	Подробные метрики для ResNet, 5 классов	195
36	Подробные метрики для ResNeXt-50, 5 классов	196
37	Подробные метрики для SE-DenseNet-121, 5 классов	196
38	Подробные метрики для SE-Inception, 5 классов	196
39	Подробные метрики для SE-ResNet, 5 классов	197
40	Подробные метрики для SE-ResNeXt-50, 5 классов	198
41	Подробные метрики для SE-Xception, 5 классов	198
42	Подробные метрики для Xception, 5 классов	198
43	Подробные метрики для DenseNet-121, 4 класса	199
44	Подробные метрики для Inception, 4 класса	199
45	Подробные метрики для InceptionResNetV2, 4 класса	199
46	Подробные метрики для ResNet, 4 класса	200
47	Подробные метрики для ResNeXt-50, 4 класса	200
48	Подробные метрики для SE-DenseNet-121, 4 класса	201
49	Подробные метрики для SE-Inception, 4 класса	201
50	Подробные метрики для SE-ResNeXt-50, 4 класса	201
51	Подробные метрики для SE-ResNet, 4 класса	202
52	Подробные метрики для SE-Xception, 4 класса	202
53	Подробные метрики для Xception, 4 класса	203

54	Подробные метрики для ResNet, 2 класса	203
55	Подробные метрики для DenseNet-121, 2 класса	204
56	Подробные метрики для Inception, 2 класса	204
57	Подробные метрики для ResNeXt-50, 2 класса	204
58	Подробные метрики для Xception, 2 класса	204

Словарь терминов

Аугментация: Увеличение количества данных датасета путем применения различного рода преобразований над имеющимися данными (использование случайного изменения яркости, контрастности изображения, различные аффинные преобразования, добавление шума и т.д.)

Батч (в нейросетях): Подмножество исходного датасета, которое «пропускается» через нейронную сеть за один проход

Пуллинг: Особый тип сверточного слоя в нейронной сети, в котором вместо весов каждого пикселя внутри скользящего окна используется некоторая заранее определенная статическая функция — выбор максимального элемента, среднее значение и т.п.

Приложение А

Структура сверточных нейронных сетей различных архитектур

В данном приложении приведено подробное описание основных архитектур сверточных нейронных сетей в виде таблиц. При этом используются сокращения для обозначения различных строительных блоков: **conv** обозначает сверточный слой с нормализацией батчей, **max pool** — слой пуллинга, **average pool** — глобальный усредняющий пуллинг, **fc** — полносвязный слой соответствующей размерности, а **se** $[r, N]$ — SE-блок с коэффициентом редукции r и выходным количеством слоев N .

Таблица 26 — Структура сверточных нейронных сетей с архитектурой ResNet различной глубины

Тип	Выход	ResNet-18	ResNet-34	ResNet-50
conv1	112×112	7×7 conv, 64, stride 2		
conv2_x	56×56	3×3 max pool, stride 2		
		$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	3×3 max pool, stride 2		
		$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$
conv4_x	14×14	3×3 max pool, stride 2		
		$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$
conv5_x	7×7	3×3 max pool, stride 2		
		$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax		

Таблица 27 — Структура сети с архитектурой ResNeXt, C — кардинальность соответствующего блока

Тип	Выход	ResNeXt-50
conv1	112×112	7×7 conv, 64, stride 2
conv2_x	56×56	3×3 max pool, stride 2
		$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128, C = 32 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	3×3 max pool, stride 2
		$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256, C = 32 \\ 1 \times 1, 512 \end{bmatrix} \times 4$
conv4_x	14×14	3×3 max pool, stride 2
		$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512, C = 32 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$
conv5_x	7×7	3×3 max pool, stride 2
		$\begin{bmatrix} 1 \times 1, 1024 \\ 3 \times 3, 1024, C = 32 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	7×7 global average pooling
		1000D fully-connected, softmax

DenseNet и особенности реализации плотно-связанных блоков. Как можно увидеть из таблицы 28, плотно-связанный блок состоит из серии пар сверток 1×1 и 3×3 . Свертка 1×1 принимает на вход входные данные блока и набор данных из всех предыдущих слоев, т.е. k -й слой в таком блоке получает на вход $N + k * 32$ каналов, где N — размерность входных данных блока, и на выходе формирует данные из 128 каналов, которые сверткой 3×3 затем уменьшаются до 32-х и подаются, наряду с входными данными блока и данными из всех предыдущих слоев, на вход следующим слоям. Таким образом, свертка 1×1 позволяет уменьшить количество карт признаков и улучшить вычислительную сложность, а свертка 3×3 обрабатывает межпространственную корреляцию между слоями.

На выходе плотно-связанного блока мы имеем данные той же самой пространственной размерности и количеством каналов, равным $N + M * 32$, где M — количество плотных слоев.

Таблица 28 — Структура сети с архитектурой DenseNet-121. Каждый сверточный слой плотно-связанного (Dense Block) или переходного (Transition) при этом сопровождается использованием нормализации батчей и нелинейностью в виде ReLU

Тип	Выход	DenseNet-121
Convolution	112×112	$7 \times 7, 64$ conv, шаг 2
Pooling	56×56	$3 \times 3, 64$ max pool, шаг 2
Dense Block (1)	56×56	$\begin{bmatrix} 1 \times 1, 128 \text{ conv} \\ 3 \times 3, 32 \text{ conv} \end{bmatrix} \times 6$, 256
Transition (1)	56×56	1×1 conv, 128, шаг 2
	28×28	2×2 avg pool, шаг 2
Dense Block (2)	28×28	$\begin{bmatrix} 1 \times 1, 128 \text{ conv} \\ 3 \times 3, 32 \text{ conv} \end{bmatrix} \times 12$, 512
Transition (2)	28×28	1×1 conv, 256, шаг 2
	14×14	2×2 avg pool, шаг 2
Dense Block (3)	14×14	$\begin{bmatrix} 1 \times 1, 128 \text{ conv} \\ 3 \times 3, 32 \text{ conv} \end{bmatrix} \times 24$, 1024
Transition (3)	14×14	1×1 conv, 512, шаг 2
	7×7	2×2 avg pool, шаг 2
Dense Block (4)	7×7	$\begin{bmatrix} 1 \times 1, 128 \text{ conv} \\ 3 \times 3, 32 \text{ conv} \end{bmatrix} \times 16$, 1024
Classification	1×1	7×7 global average pooling
		1000D fully-connected, softmax

Интеграция блоков сжатия и возбуждения. Структура сетей ResNet и ResNeXt различной глубины с интегрированными SE-блоками, принимающих на вход изображение размером 224×224 , представлена в таблице 29. Каждый сверточный слой сопровождается нормализацией батчей, а первый слой базового блока (SE-ResNet-18, SE-ResNet-34) и первые два слоя для расширенного блока (SE-ResNet-50) — еще и нелинейностью в виде ReLU, значение кардинальности $C = 32$.

Структура сети с архитектурой SE-DenseNet-121, показана в таблице 30. Здесь $se[r]$ — SE-блок с коэффициентом редукции r и соответствующим своему положению количеством входных и выходных каналов (зависит от плотно-связанного слоя и т.п.).

Таблица 29 — Структура сетей ResNet и ResNeXt различной глубины с интегрированными SE-блоками

Выход	SE-ResNet-18	SE-ResNet-34	SE-ResNet-50	SE-ResNeXt-50
112×112	7×7 conv, 64, шаг 2			
56×56	3×3 max pool, шаг 2			
	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \\ \text{se}[16, 64] \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \\ \text{se}[16, 64] \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \\ \text{se}[16, 256] \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128, C \\ 1 \times 1, 256 \\ \text{se}[16, 256] \end{bmatrix} \times 3$
28×28	3×3 max pool, шаг 2			
	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \\ \text{se}[16, 128] \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \\ \text{se}[16, 128] \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \\ \text{se}[16, 512] \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256, C \\ 1 \times 1, 512 \\ \text{se}[32, 512] \end{bmatrix} \times 4$
14×14	3×3 max pool, шаг 2			
	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \\ \text{se}[16, 256] \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \\ \text{se}[16, 256] \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \\ \text{se}[16, 1024] \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512, C \\ 1 \times 1, 1024 \\ \text{se}[64, 1024] \end{bmatrix} \times 6$
7×7	3×3 max pool, шаг 2			
	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \\ \text{se}[16, 512] \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \\ \text{se}[16, 512] \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \\ \text{se}[16, 2048] \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 1024 \\ 3 \times 3, 1024, C \\ 1 \times 1, 2048 \\ \text{se}[128, 2048] \end{bmatrix} \times 3$
1×1	average pool, 1000-d fc, softmax			

Таблица 30 — Структура сети с архитектурой SE-DenseNet-121

Тип	Выход	DenseNet-121
Convolution	112×112	7×7 conv, 64, stride 2
Pooling	56×56	3×3 max pool, stride 2
SE-Block	56×56	se[4, 64]
Dense Block (1)	56×56	se[4, 64]
		$\begin{bmatrix} \text{se}[16] \\ 1 \times 1 \text{ conv, } 128 \\ 3 \times 3 \text{ conv, } 32 \end{bmatrix} \times 6, 256$
Transition (1)	56×56	se[16, 256]
		1×1 conv, 128, stride 2
	28×28	2×2 avg pool, stride 2
Dense Block (2)	28×28	se[8, 128]
		$\begin{bmatrix} \text{se}[16] \\ 1 \times 1 \text{ conv, } 128 \\ 3 \times 3 \text{ conv, } 32 \end{bmatrix} \times 12, 512$
Transition (2)	28×28	se[32, 512]
		1×1 conv, 256, stride 2
	14×14	2×2 avg pool, stride 2
Dense Block (3)	14×14	se[16, 256]
		$\begin{bmatrix} \text{se}[16] \\ 1 \times 1 \text{ conv, } 128 \\ 3 \times 3 \text{ conv, } 32 \end{bmatrix} \times 24, 1024$
Transition (3)	14×14	se[64, 1024]
		1×1 conv, 512, stride 2
	7×7	2×2 avg pool, stride 2
Dense Block (4)	7×7	se[32, 512]
		$\begin{bmatrix} \text{se}[16] \\ 1 \times 1 \text{ conv, } 128 \\ 3 \times 3 \text{ conv, } 32 \end{bmatrix} \times 16, 1024$
Classification	1×1	7×7 global average pooling
		1000D fully-connected, softmax

Приложение Б

Таблицы со значениями метрик классификации для традиционных архитектур сверточных нейронных сетей

Таблица 31 — Подробные метрики для DenseNet-121, 5 классов

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,715	0,826	0,766	0,728	0,802	0,763
1	0,330	0,208	0,255	0,355	0,330	0,336
2	0,643	0,669	0,655	0,677	0,626	0,649
3	0,804	0,801	0,802	0,813	0,773	0,791
4	0,876	0,745	0,801	0,892	0,817	0,850
Среднее	0,644	0,667	0,651	0,664	0,667	0,662
Макс.	0,662	0,681	0,669	0,673	0,678	0,666

Таблица 32 — Подробные метрики для Inception, 5 классов

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,729	0,866	0,791	0,735	0,870	0,796
1	0,392	0,223	0,282	0,383	0,331	0,352
2	0,683	0,707	0,695	0,729	0,619	0,669
3	0,837	0,813	0,825	0,809	0,809	0,808
4	0,884	0,810	0,844	0,920	0,654	0,756
Среднее	0,676	0,699	0,680	0,686	0,691	0,683
Макс.	0,686	0,707	0,693	0,695	0,695	0,690

Таблица 33 — Подробные метрики для InceptionResNetV2, 5 классов

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,747	0,868	0,802	0,707	0,895	0,790
1	0,364	0,203	0,244	0,341	0,236	0,269
2	0,679	0,738	0,706	0,715	0,573	0,632
3	0,796	0,815	0,800	0,758	0,813	0,784
4	0,869	0,791	0,825	0,904	0,719	0,792
Среднее	0,671	0,704	0,677	0,657	0,674	0,653
Макс.	0,709	0,721	0,710	0,670	0,682	0,666

Таблица 34 — Подробные метрики для InceptionResNetV2, 2 класса

ОА	Precision	Recall	F1
0	0,908	0,762	0,828
1	0,837	0,940	0,885
Среднее.:	0,868	0,862	0,860
Макс.:	0,871	0,865	0,864

Таблица 35 — Подробные метрики для ResNet, 5 классов

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
ResNet-18						
0	0,682	0,854	0,758	0,727	0,826	0,773
1	0,281	0,146	0,188	0,323	0,277	0,2981
2	0,635	0,618	0,620	0,674	0,631	0,651
3	0,796	0,776	0,783	0,813	0,776	0,793
4	0,894	0,817	0,853	0,902	0,771	0,822
Среднее	0,619	0,652	0,625	0,658	0,667	0,660
Макс.	0,623	0,664	0,627	0,663	0,673	0,664
ResNet-34						
0	0,687	0,865	0,763	0,713	0,808	0,757
1	0,318	0,222	0,248	0,294	0,258	0,273
2	0,715	0,589	0,646	0,668	0,632	0,650
3	0,797	0,831	0,813	0,834	0,767	0,798
4	0,902	0,765	0,823	0,910	0,771	0,834
Среднее	0,650	0,668	0,648	0,649	0,656	0,649
Макс.	0,664	0,673	0,662	0,650	0,663	0,653
ResNet-50						
0	0,698	0,855	0,768	0,718	0,809	0,760
1	0,343	0,200	0,250	0,345	0,259	0,2961
2	0,642	0,645	0,643	0,653	0,644	0,648
3	0,814	0,723	0,766	0,770	0,789	0,777
4	0,877	0,876	0,874	0,921	0,758	0,832
Среднее	0,640	0,664	0,645	0,647	0,662	0,651
Макс.	0,646	0,675	0,651	0,661	0,677	0,664

Таблица 36 — Подробные метрики для ResNeXt-50, 5 классов

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,718	0,855	0,780	0,747	0,814	0,779
1	0,342	0,200	0,252	0,331	0,316	0,321
2	0,651	0,665	0,657	0,675	0,619	0,642
3	0,792	0,779	0,785	0,795	0,762	0,776
4	0,934	0,752	0,832	0,801	0,752	0,775
Среднее	0,650	0,673	0,654	0,662	0,663	0,660
Макс.	0,674	0,695	0,677	0,678	0,666	0,664

Таблица 37 — Подробные метрики для SE-DenseNet-121, 5 классов

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,733	0,859	0,790	0,713	0,879	0,787
1	0,417	0,261	0,315	0,362	0,265	0,302
2	0,702	0,688	0,694	0,723	0,650	0,684
3	0,812	0,833	0,821	0,875	0,777	0,819
4	0,828	0,784	0,805	0,865	0,837	0,850
Среднее	0,682	0,700	0,684	0,680	0,692	0,679
Макс.	0,693	0,717	0,695	0,688	0,702	0,684

Таблица 38 — Подробные метрики для SE-Inception, 5 классов

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,726	0,879	0,795	0,738	0,845	0,788
1	0,372	0,205	0,259	0,346	0,329	0,337
2	0,690	0,677	0,681	0,717	0,625	0,668
3	0,810	0,834	0,821	0,833	0,782	0,806
4	0,856	0,810	0,832	0,875	0,752	0,803
Среднее	0,669	0,696	0,673	0,679	0,682	0,678
Макс.	0,671	0,702	0,676	0,688	0,688	0,684

Таблица 39 — Подробные метрики для SE-ResNet, 5 классов

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
SE-ResNet-18						
0	0,671	0,899	0,768	0,713	0,796	0,751
1	0,342	0,207	0,254	0,287	0,259	0,272
2	0,680	0,567	0,617	0,642	0,600	0,621
3	0,820	0,732	0,773	0,812	0,765	0,783
4	0,896	0,797	0,841	0,898	0,765	0,822
Среднее	0,642	0,660	0,638	0,644	0,649	0,644
Макс.	0,652	0,669	0,644	0,637	0,642	0,636
SE-ResNet-34						
0	0,699	0,870	0,775	0,719	0,827	0,769
1	0,339	0,216	0,256	0,307	0,249	0,271
2	0,671	0,614	0,639	0,685	0,632	0,656
3	0,828	0,801	0,813	0,806	0,807	0,806
4	0,877	0,752	0,800	0,902	0,771	0,830
Среднее	0,650	0,671	0,651	0,653	0,667	0,656
Макс.	0,656	0,684	0,657	0,660	0,679	0,658
SE-ResNet-50						
0	0,696	0,908	0,786	0,753	0,798	0,775
1	0,346	0,153	0,209	0,358	0,389	0,372
2	0,702	0,675	0,686	0,684	0,626	0,654
3	0,833	0,776	0,799	0,845	0,750	0,793
4	0,863	0,824	0,841	0,872	0,804	0,828
Среднее	0,659	0,690	0,659	0,680	0,672	0,674
Макс.	0,668	0,691	0,665	0,689	0,690	0,688

Таблица 40 — Подробные метрики для SE-ResNeXt-50, 5 классов

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,731	0,882	0,799	0,759	0,849	0,799
1	0,408	0,200	0,267	0,390	0,345	0,362
2	0,700	0,746	0,722	0,712	0,674	0,692
3	0,835	0,818	0,826	0,853	0,759	0,801
4	0,898	0,784	0,835	0,830	0,804	0,808
Среднее	0,684	0,712	0,688	0,696	0,698	0,693
Макс.	0,697	0,719	0,701	0,702	0,707	0,699

Таблица 41 — Подробные метрики для SE-Xception, 5 классов

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,732	0,892	0,803	0,730	0,886	0,800
1	0,386	0,283	0,325	0,382	0,270	0,315
2	0,715	0,623	0,664	0,671	0,696	0,683
3	0,796	0,779	0,787	0,885	0,641	0,740
4	0,858	0,765	0,786	0,883	0,778	0,826
Среднее	0,678	0,691	0,677	0,678	0,688	0,674
Макс.	0,694	0,703	0,694	0,693	0,704	0,693

Таблица 42 — Подробные метрики для Xception, 5 классов

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,720	0,895	0,797	0,740	0,847	0,789
1	0,351	0,214	0,240	0,359	0,295	0,320
2	0,687	0,699	0,689	0,706	0,685	0,692
3	0,897	0,707	0,790	0,869	0,755	0,807
4	0,838	0,804	0,812	0,837	0,830	0,833
Среднее	0,672	0,692	0,668	0,683	0,692	0,683
Макс.	0,698	0,700	0,687	0,696	0,699	0,691

Таблица 43 — Подробные метрики для DenseNet-121, 4 класса

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,840	0,925	0,880	0,826	0,940	0,879
1	0,710	0,585	0,641	0,729	0,533	0,613
2	0,819	0,779	0,797	0,816	0,786	0,800
3	0,874	0,771	0,819	0,880	0,810	0,844
Среднее	0,803	0,809	0,802	0,800	0,806	0,796
Макс.	0,810	0,816	0,810	0,805	0,812	0,802

Таблица 44 — Подробные метрики для Inception, 4 класса

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,847	0,938	0,890	0,834	0,923	0,876
1	0,747	0,617	0,676	0,680	0,612	0,643
2	0,864	0,759	0,806	0,906	0,647	0,752
3	0,810	0,850	0,828	0,806	0,889	0,843
Среднее	0,821	0,825	0,819	0,801	0,801	0,796
Макс.	0,829	0,832	0,826	0,815	0,817	0,810

Таблица 45 — Подробные метрики для InceptionResNetV2, 4 класса

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,835	0,932	0,881	0,844	0,928	0,883
1	0,725	0,572	0,639	0,739	0,577	0,645
2	0,797	0,782	0,788	0,801	0,807	0,803
3	0,910	0,712	0,796	0,840	0,804	0,820
Среднее	0,803	0,808	0,800	0,810	0,813	0,806
Макс.	0,810	0,814	0,806	0,813	0,818	0,813

Таблица 46 — Подробные метрики для ResNet, 4 класса

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
ResNet-18						
0	0,824	0,923	0,870	0,819	0,930	0,871
1	0,710	0,528	0,602	0,695	0,527	0,599
2	0,789	0,809	0,798	0,821	0,761	0,788
3	0,898	0,778	0,832	0,866	0,797	0,830
Среднее	0,791	0,796	0,787	0,787	0,795	0,785
Макс.	0,792	0,798	0,794	0,789	0,796	0,789
ResNet-34						
0	0,823	0,916	0,867	0,837	0,914	0,873
1	0,698	0,539	0,606	0,695	0,594	0,635
2	0,796	0,788	0,792	0,856	0,735	0,790
3	0,882	0,810	0,844	0,825	0,882	0,852
Среднее	0,788	0,794	0,786	0,801	0,803	0,797
Макс.	0,795	0,801	0,796	0,813	0,808	0,809
ResNet-50						
0	0,832	0,931	0,878	0,844	0,913	0,877
1	0,720	0,556	0,626	0,704	0,579	0,635
2	0,793	0,762	0,777	0,780	0,795	0,787
3	0,883	0,810	0,841	0,871	0,817	0,839
Среднее	0,798	0,804	0,796	0,798	0,804	0,798
Макс.	0,801	0,807	0,798	0,800	0,807	0,799

Таблица 47 — Подробные метрики для ResNeXt-50, 4 класса

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,824	0,936	0,876	0,825	0,923	0,871
1	0,742	0,544	0,626	0,697	0,566	0,620
2	0,810	0,774	0,790	0,776	0,723	0,749
3	0,825	0,824	0,823	0,562	0,529	0,545
Среднее	0,800	0,805	0,795	0,776	0,788	0,777
Макс.	0,809	0,812	0,805	0,804	0,809	0,801

Таблица 48 — Подробные метрики для SE-DenseNet-121, 4 класса

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,867	0,906	0,886	0,853	0,923	0,886
1	0,723	0,672	0,695	0,718	0,646	0,679
2	0,827	0,804	0,814	0,877	0,759	0,814
3	0,886	0,771	0,820	0,844	0,797	0,817
Среднее	0,824	0,825	0,823	0,820	0,822	0,818
Макс.	0,830	0,833	0,828	0,830	0,832	0,830

Таблица 49 — Подробные метрики для SE-Inception, 4 класса

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,827	0,939	0,880	0,849	0,924	0,885
1	0,735	0,588	0,652	0,730	0,613	0,664
2	0,869	0,750	0,805	0,827	0,815	0,819
3	0,896	0,745	0,805	0,897	0,712	0,791
Среднее	0,810	0,813	0,806	0,815	0,819	0,813
Макс.	0,811	0,813	0,808	0,825	0,824	0,824

Таблица 50 — Подробные метрики для SE-ResNeXt-50, 4 класса

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,842	0,919	0,878	0,851	0,920	0,884
1	0,725	0,594	0,648	0,703	0,641	0,670
2	0,822	0,783	0,802	0,857	0,729	0,787
3	0,835	0,830	0,831	0,878	0,804	0,837
Среднее	0,807	0,810	0,804	0,813	0,815	0,812
Макс.	0,816	0,820	0,817	0,815	0,819	0,816

Таблица 51 — Подробные метрики для SE-ResNet, 4 класса

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
SE-ResNet-18						
0	0,826	0,927	0,873	0,855	0,876	0,864
1	0,704	0,535	0,607	0,654	0,632	0,641
2	0,803	0,774	0,788	0,809	0,789	0,798
3	0,848	0,830	0,839	0,882	0,784	0,830
Среднее	0,790	0,798	0,789	0,795	0,795	0,794
Макс.	0,798	0,804	0,799	0,802	0,797	0,798
SE-ResNet-34						
0	0,826	0,927	0,874	0,847	0,898	0,872
1	0,719	0,546	0,619	0,686	0,594	0,636
2	0,806	0,807	0,805	0,784	0,812	0,796
3	0,901	0,765	0,827	0,900	0,810	0,849
Среднее	0,797	0,803	0,794	0,797	0,801	0,797
Макс.	0,800	0,806	0,799	0,810	0,814	0,811
SE-ResNet-50						
0	0,841	0,905	0,871	0,834	0,922	0,874
1	0,682	0,610	0,643	0,709	0,579	0,630
2	0,852	0,738	0,791	0,844	0,753	0,795
3	0,849	0,908	0,877	0,843	0,824	0,831
Среднее	0,799	0,803	0,799	0,802	0,804	0,796
Макс.	0,806	0,807	0,806	0,811	0,813	0,812

Таблица 52 — Подробные метрики для SE-Xception, 4 класса

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,825	0,948	0,882	0,850	0,911	0,879
1	0,749	0,558	0,638	0,699	0,637	0,665
2	0,855	0,782	0,816	0,857	0,765	0,807
3	0,904	0,771	0,823	0,888	0,784	0,827
Среднее	0,811	0,815	0,805	0,811	0,813	0,810
Макс.	0,819	0,824	0,819	0,816	0,815	0,813

Таблица 53 — Подробные метрики для Xception, 4 класса

ОА	Cross-Entropy Loss			Ordinal Loss		
	Precision	Recall	F1	Precision	Recall	F1
0	0,856	0,904	0,879	0,853	0,911	0,881
1	0,703	0,646	0,671	0,692	0,628	0,658
2	0,837	0,795	0,815	0,815	0,761	0,783
3	0,898	0,745	0,805	0,914	0,758	0,828
Среднее	0,813	0,815	0,812	0,806	0,809	0,806
Макс.	0,818	0,819	0,817	0,810	0,812	0,810

Таблица 54 — Подробные метрики для ResNet, 2 класса

ОА	ResNet			SE-ResNet		
	Precision	Recall	F1	Precision	Recall	F1
ResNet-18						
0	0,840	0,923	0,879	0,837	0,904	0,869
1	0,886	0,771	0,824	0,861	0,771	0,813
Среднее	0,860	0,857	0,856	0,847	0,846	0,844
Макс.	0,861	0,859	0,858	0,856	0,856	0,856
ResNet-34						
0	0,843	0,902	0,871	0,835	0,925	0,877
1	0,861	0,783	0,820	0,887	0,762	0,820
Среднее	0,851	0,850	0,849	0,857	0,854	0,852
Макс.	0,865	0,862	0,861	0,860	0,856	0,855
ResNet-50						
0	0,837	0,912	0,873	0,836	0,918	0,875
1	0,871	0,770	0,817	0,878	0,766	0,818
Среднее	0,852	0,850	0,849	0,854	0,852	0,850
Макс.	0,860	0,857	0,856	0,861	0,859	0,857

Таблица 55 — Подробные метрики для DenseNet-121, 2 класса

ОА	DenseNet121			SE-DenseNet121		
	Precision	Recall	F1	Precision	Recall	F1
0	0,823	0,929	0,873	0,854	0,913	0,882
1	0,890	0,742	0,809	0,876	0,798	0,835
Среднее	0,853	0,848	0,845	0,864	0,863	0,862
Макс.	0,858	0,853	0,850	0,866	0,865	0,865

Таблица 56 — Подробные метрики для Inception, 2 класса

ОА	Inception			SE-Inception		
	Precision	Recall	F1	Precision	Recall	F1
0	0,930	0,723	0,813	0,877	0,789	0,831
1	0,818	0,958	0,882	0,849	0,914	0,881
Среднее	0,867	0,856	0,852	0,861	0,860	0,859
Макс.	0,873	0,862	0,859	0,865	0,863	0,862

Таблица 57 — Подробные метрики для ResNeXt-50, 2 класса

ОА	ResNeXt50			SE-ResNeXt50		
	Precision	Recall	F1	Precision	Recall	F1
0	0,818	0,935	0,873	0,877	0,798	0,836
1	0,898	0,730	0,805	0,855	0,913	0,883
Среднее	0,853	0,846	0,843	0,865	0,863	0,862
Макс.	0,855	0,853	0,852	0,866	0,866	0,866

Таблица 58 — Подробные метрики для Xception, 2 класса

ОА	Xception			SE-Xception		
	Precision	Recall	F1	Precision	Recall	F1
0	0,899	0,773	0,831	0,877	0,808	0,839
1	0,842	0,933	0,885	0,861	0,911	0,885
Среднее	0,867	0,863	0,861	0,868	0,866	0,865
Макс.	0,870	0,865	0,864	0,875	0,875	0,875

Приложение В

**Свидетельство о государственной регистрации программы для ЭВМ
№2022665581**

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2022665581

**Программа для автоматической диагностики
остеоартрита коленного сустава по
рентгенографическому изображению**

Правообладатель: *Михайличенко Алексей Андреевич (RU)*

Автор(ы): *Михайличенко Алексей Андреевич (RU)*



Заявка № 2022664895

Дата поступления 04 августа 2022 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 18 августа 2022 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Ю.С. Зубов

Приложение Г

Акты о внедрении результатов работы

УТВЕРЖДАЮ

Директор

Института математики, механики
и компьютерных наук им.

И.И.Ворожича

Южного федерального
университета

М.И. Карякин

2022 г.

АКТ

об использовании результатов диссертации

Михайличенко Алексея Андреевича

в учебном процессе кафедры прикладной математики и информатики
Института математики, механики и компьютерных наук Южного
федерального университета

Комиссия в составе:

председателя Ревинной С.В.

И членов комиссии Гуды С.А., Михалковича С.С., Угольницкого Г.А.

Составили настоящий акт о том, что результаты диссертационной работы

**«Методы и алгоритмы автоматизированной диагностики
остеоартрита по рентгенографическим изображениям»,**

представленной на соискание ученой степени кандидата технических наук,
использованы в учебном процессе кафедры прикладной математики и
программирования (ПМП) Института математики, механики и
компьютерных наук (ММиКН) Южного федерального университета
(ЮФУ).

Настоящий акт подтверждает, что в учебном процессе кафедры ПМП
ММиКН ЮФУ используются следующие научно-теоретические и
практические результаты кандидатской диссертации Михайличенко А.А.:

- метод локализации коленного сустава на изображении при помощи архитектуры Single Shot Detector (SSD), включая исходный код на языке Python;

- метод автоматической классификации стадии остеоартрита коленного сустава по изображению при помощи свёрточных нейронных сетей, включая исходный код процесса обучения, тестирования и анализа результатов классификации на языке Python;
- программа, демонстрирующая работу представленных в диссертации методов (свидетельство о государственной регистрации программы для ЭВМ №2022665581).

Указанные материалы используются в лекционных курсах по дисциплинам «Компьютерное зрение», «Машинное обучение», «Нейронные сети для мобильных приложений» (для подготовки магистров направления подготовки 02.04.02 Фундаментальная информатика и информационные технологии (магистерская образовательная программа «Разработка мобильных приложений и компьютерных игр»)).

Председатель комиссии:

Зам директора по развитию магистратуры
ИММиКН,

доцент ИММиКН ЮФУ,
к.ф.-м.н., доцент



С.В. Ревина

Члены комиссии:

Лектор дисциплины

«Машинное обучение»,
доцент ИММиКН ЮФУ,
к.т.н., доцент



С.А. Гуда

Зав. каф. Информатики и
вычислительного эксперимента
ИММиКН ЮФУ,
к.ф.-м.н., доцент



С.С. Михалкович

Зав. каф. прикладной математики и
программирования ИММиКН ЮФУ
д.ф.м.н., проф.



Г.А. Угольницкий

Общество с ограниченной
ответственностью «ИЗОСКАН»
ИНН 6194001668, ОГРН 1166196098892,
344001, г. Ростов-на-Дону,
ул. Некрасовская, 75, литер А, этаж 2, к. 6
info@isoscan.ru, тел: +7 (863) 275-95-05

УТВЕРЖДАЮ

Директор

ООО «ИЗОСКАН» к.ф.-м.н.

А.Б. Клещенко



2022 г.

АКТ

о внедрении результатов кандидатской диссертации
«Методы и алгоритмы автоматизированной диагностики остеоартрита по
рентгенографическим изображениям»
Михайличенко Алексея Андреевича

Комиссия в составе председателя, директора ООО «ИЗОСКАН», к.ф.-м.н. Клещенко А.Б., главного инженера Египко В.А. и начальника отдела, к.ф.-м.н. Крутиева С.В., составила настоящий акт о том, что в разработках ООО «ИЗОСКАН» внедрены следующие результаты диссертации Михайличенко Алексея Андреевича на тему «Методы и алгоритмы автоматизированной диагностики остеоартрита по рентгенографическим изображениям»:

1. Алгоритм локализации объектов при помощи модели сверточной сети с архитектурой Single Shot Detector.
2. Алгоритм автоматической классификации объектов на несколько классов по графическим данным на базе представленного в диссертации метода автоматической классификации остеоартрита коленного сустава по рентгенограмме при помощи сверточных нейронных сетей.
3. Вычислительные структуры и исходный код алгоритмов локализации и классификации на языке C++, а также код обучения и тестирования сверточных нейронных сетей, используемых для решения задачи классификации, на языке Python.

Указанные результаты диссертации Михайличенко А.А. были внедрены при реализации программного комплекса, решающего задачу автоматической классификации объектов по графическим данным (изображениям).

Внедрение результатов исследования диссертации Михайличенко А.А. позволило реализовать эффективный алгоритм классификации, точность работы которого превышает 90%.

ПРЕДСЕДАТЕЛЬ КОМИССИИ:

директор ООО «ИЗОСКАН»,

к.ф.-м.н.

 А.Б. Клещенко

ЧЛЕНЫ КОМИССИИ:

Главный инженер ООО «ИЗОСКАН»

 В.А. Египко

Начальник отдела

к.ф.-м.н.

 С.В. Крутиев



Общество с ограниченной ответственностью «Наука»

ИНН: 6162078107, ОГРН: 1186196008283,

КПП 616201001

Лицензия МЗ РО № ЛО-61-01-007733 от 05.06.2020

344034, г. Ростов-на-Дону, ул. Загорская, 23 А

222-85-35 222-85-45

УТВЕРЖДАЮ

Заместитель генерального директора

ООО «Наука»

Челохов Ю.К.

Исх. № 1128 от « 02 » 12 2022 г.

12 2022 г.



АКТ

о внедрении результатов кандидатской диссертации
младшего научного сотрудника НИТЦ нейротехнологий ЮФУ
Михайличенко Алексея Андреевича

Комиссия в составе председателя д.б.н., профессора Шкурат Т.П., и членов комиссии: заведующей клинико-диагностической лабораторией, к.б.н. Александровой А.А., биолога клинико-диагностической лаборатории, к.б.н. Мельниковой Е.С., составила настоящий акт о том, что в разработках ООО «Наука» внедрены следующие результаты диссертации Михайличенко Алексея Андреевича на тему «Методы и алгоритмы автоматизированной диагностики остеоартрита по рентгенографическим изображениям»:

1. Алгоритм автоматической классификации объекта по изображению при помощи сверточных нейронных сетей, основанный на описанном в диссертации алгоритме классификации остеоартрита коленного сустава по рентгенографическим данным.
2. Готовые модели сверточных нейронных сетей различных архитектур на языке Python с использованием фреймворка pytorch и исходный код обучения и тестирования моделей.

Указанные наработки, полученные в результате написания диссертации Михайличенко А.А., были использованы для создания автоматического алгоритма классификации объектов на несколько классов по изображению этих объектов и имеющейся дополнительной информации. В процессе реализации алгоритма использовались готовые структуры моделей сверточных нейронных сетей, созданные в рамках проведенных исследований, а также исходный код обучения моделей, написанный на языке Python, и модуль преобразования полученных моделей в формат, пригодный для использования в других языках (в частности, C/C++).

В результате внедрения имеющихся наработок, был реализован алгоритм классификации, работающий как на графических ускорителях, так и на центральном

процессоре, при этом точность классификации алгоритма составила 93,74%, что является достаточным для практического использования в целевой области.

ПРЕДСЕДАТЕЛЬ КОМИССИИ:

Д.б.н., профессор



Шкурят Т.П.

ЧЛЕНЫ КОМИССИИ:

Заведующая лабораторией, к.б.н.



Александрова А.А.

Биолог, к.б.н.



Мельникова Е.С.

Приложение Д

Описание и краткая инструкция к разработанному программному обеспечению

В ходе проведения исследования было выполнено обучение множества моделей сверточных нейронных сетей различных архитектур, предназначенных для решения задачи классификации остеоартрита коленного сустава по шкале Келлгрена-Лоуренса по рентгенограмме сустава. Дополнительно было проведено обучение детектора для локализации коленного сустава на рентгенограмме, результаты работы которого используются в модулях классификации. Для возможности практического применения полученных результатов был разработан программный комплекс, который объединяет в себе полученные результаты и предоставляет возможность использовать эти результаты в медицинских целях.

Данная глава посвящена разработанному в рамках данной работы программному обеспечению. Приводятся как особенности программной реализации отдельных модулей, так и доступные пользователю ПО инструменты для проведения измерения или автоматической классификации остеоартрита на основе полученных практических результатов исследования классификаторов на базе сверточных нейронных сетей.

Д.1 Общее описание

Помимо функционала по автоматическому распознаванию степени ОА, в разработанном приложении также доступны различные инструменты для проведения медицинских измерений — оценки различных текстурных характеристик участков рентгенограммы, измерение длин и углов и т.п. Для повышения точности измерений и удобства выделения участков интереса разработан специальный инструмент *Умная кривая*, позволяющий проводить автоматическое уточнение положения расставляемых пользователем ключевых точек на основе имеющихся краев в изображении, опираясь на векторное поле потока градиента изображения (gradient vector flow field, GVF [137]).

Программный код написан на языке C++ с использованием набора библиотек **Qt**, что позволяет сделать проект кроссплатформенным, включая графический интерфейс и др. Для работы с нейросетями используется библиотека **pytorch** и ее wrapper для языка C/C++ под названием *libtorch*. Исходный код приложения и сопутствующих инструментов доступен по лицензии MIT, и может свободно использоваться и распространяться в личных и/или коммерческих целях.

Д.2 Базовый функционал

Разработанное программное обеспечение поддерживает загрузку изображений в различных графических форматах, а также загрузку данных в медицинском формате DICOM, являющимся де-факто стандартом для передачи подобного рода информации. Доступна загрузка как одиночных файлов, так и пакетная обработка (набор файлов или содержимое выбранной пользователем директории).

Пользователь имеет возможность произвольно масштабировать изображение в области просмотра (включая масштабирование под размеры рабочей области) и передвигать его.

После загрузки изображения, пользователю доступен ряд инструментов для проведения различных измерений. Внешний вид панели инструментов представлен на рис. 76. Вся панель инструментов делится на три группы — опции масштабирования, инструменты для проведения медицинских измерений и группа инструментов для преобразований и фильтрации входного изображения.



Рисунок 76 — Иллюстрация панели инструментов

Первая группа состоит из кнопок с номерами 1 и 2. Кнопка под номером 1 позволяет сбросить все пользовательские коэффициенты масштабирования и вписать изображение в рабочую область, кнопка 2 предоставляет доступ к различным командам масштабирования.

Д.2.1 Проведение измерений

Вторая группа кнопок содержит 6 различных инструментов для основных измерений в рамках проведения исследования содержимого медицинских рентгенограмм. Помимо стандартных инструментов Измерение длины (3) и Измерение угла (4) в приложении доступен инструмент для измерения угла Кобба (5), который является стандартной мерой измерения искривления позвоночника для оценки прогрессирования сколиоза (рис. 77). Углом искривления позвоночника считается угол между двумя перпендикулярами, проведенными к линиям, параллельным верхней границе верхнего и нижней границе нижнего позвонков. Измерение данного угла позволяет определить, необходимо ли пациенту лечение.

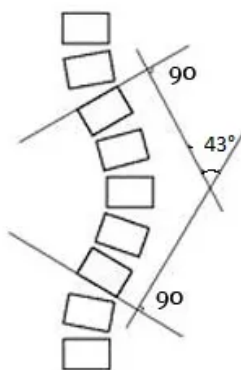


Рисунок 77 — Иллюстрация угла Кобба

Инструменты 6 и 7 позволяют выделить на изображении эллиптическую или многоугольную области, чтобы получить информацию о текстурных характеристиках желаемого участка рентгенограммы. Инструмент Умная кривая (8) служит для точного измерения длин кромки костей и использует автоматические алгоритмы соединения точек с опорой на векторное поле потока градиента изображения (GVF) для автоматической реконструкции фрагмента контура (механизм соединения ключевых точек подробно описан в работе [143]).

Внешний вид графических элементов для описанных инструментов приведен на рис. 78. Стоит отметить, что все созданные элементы являются редактируемыми и могут быть изменены (откорректированы) после их создания при помощи интуитивно понятных действий. Ниже представлена краткая пользовательская инструкция по использованию доступных инструментов.

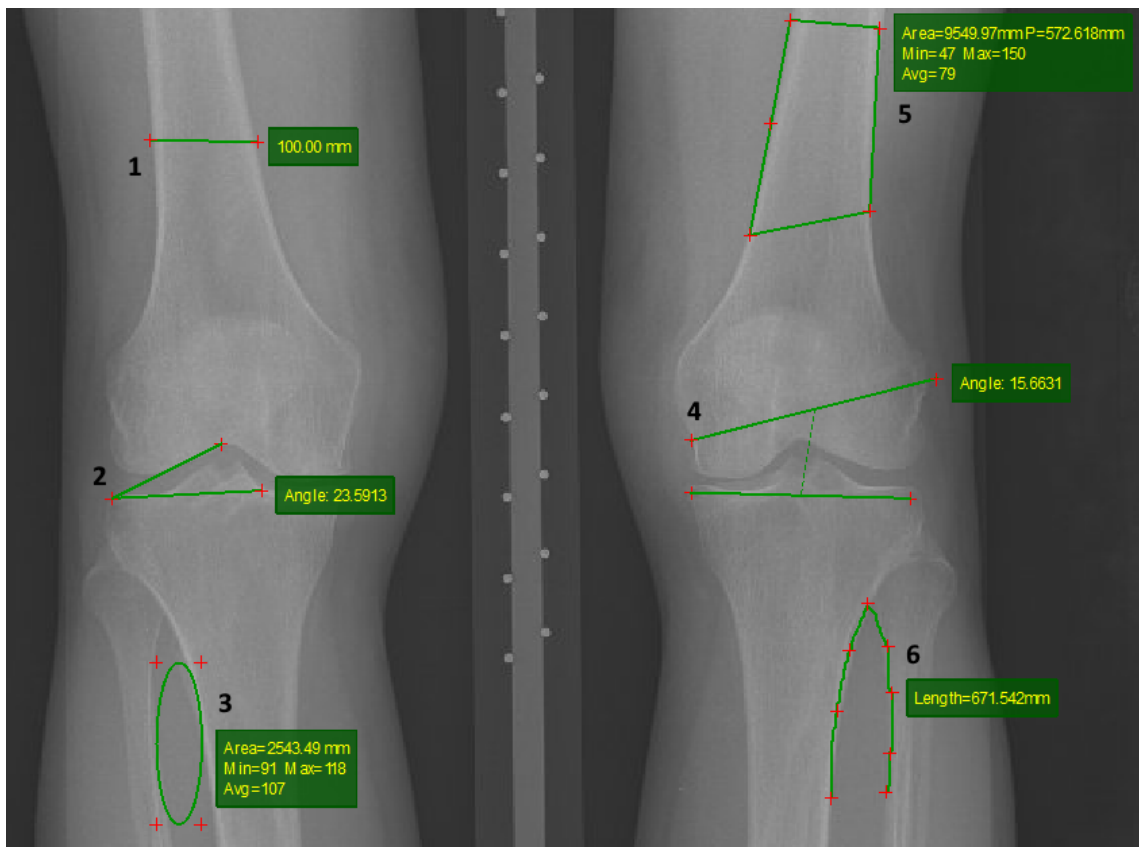


Рисунок 78 — Иллюстрация измерительных инструментов, доступных в приложении: измерение длины (1), измерение угла (2), эллиптическая область (3), угол Кобба (4), многоугольная область (5) и умная кривая (6)

- **Измерение длины.** Инструмент измерения длины размещается при помощи двух точек на изображении, выбрав соответствующий режим в панели инструментов. Пара точек может быть размещена как парой независимых кликов по изображению, так и при помощи комбинации *нажатие* (для расстановки первой точки) - *перемещения на заданную длину* - *отпускание* (для установки второй точки).
- **Измерение стандартных углов.** Угол на изображении может быть определен при помощи трех кликов для размещения трех точек угла после выбора соответствующего пункта на панели инструментов.
- **Измерение углов Кобба.** Для создания данного графического элемента в рабочей области программы необходимо определить два отрезка, формирующих открытый угол Кобба, выбрав соответствующий элемент на панель инструментов.
- **Эллиптическая область.** Эллиптическая область на изображении позволяет оценивать текстурные характеристики региона соответствующей формы на изображении, такие как среднее и предельные значения ин-

тенсивности в выбранном регионе, площадь области и т.п. Для создания данного элемента на изображении необходимо выбрать соответствующий пункт на панели инструментов, а затем при помощи двух кликов задать описывающий эллиптическую область прямоугольник.

- **Многоугольная область.** Область неправильной формы на изображении по аналогии с областью эллиптической формы. Каждая следующая точка создается кликом на изображении, последняя точка, позволяющая замкнуть область (соединив последнюю точку с первой), создается при помощи двойного клика.
- **Умная кривая.** Умная кривая — незамкнутый контур области изображения, который создается при помощи задания пользователем ключевых точек на изображении, после чего происходит автоматическое построение полного набора точек заданного контура с опорой на векторное поле потока градиента (GVF). Ключевые точки расставляются при помощи клика по изображению, а завершается процесс добавления точек двойным кликом, который устанавливает последнюю точку. Автоматический поиск промежуточных точек происходит после добавления каждой новой точки контура.

Элементы *Многоугольная область* и *Умная кривая*, помимо стандартного функционала редактирования, имеют возможность добавления к ним дополнительных точек путем двойного клика по желаемому участку элемента.

Калибровка единиц измерения. По умолчанию в качестве единиц измерения используются пиксели изображения. Чтобы конкретизировать единицы измерения, для каждого изображения необходимо произвести калибровку инструментов. Сделать это можно двумя способами.

Первый способ заключается в использовании инструмента *Измерение длины*. После создания измерительного элемента на изображении через контекстное меню можно привязать реальную длину в миллиметрах, которая соответствует длине графического элемента на изображении. После того, как для одного из элементов выполнена калибровка, эта калибровка будет использована и для всех остальных графических элементов, что будет сигнализировано изменением цвета этих элементов.

Второй способ доступен в меню *Измерение и аннотирование* и не требует предварительного создания каких-либо графических элементов. В пункте *Калибровка* достаточно ввести в диалоговом окне пиксельную длину и соот-

ветствующую ей реальную длину в миллиметрах, чтобы получить требуемый поправочный коэффициент для всех имеющихся или вновь созданных графических элементов.

При необходимости, калибровка может быть изменена или сброшена в меню *Измерение и аннотирование*.

Д.2.2 Редактирование изображения

Исходное изображение, загружаемое пользователем, может иметь какие-то дефекты или быть по тем или иным причинам непригодным для немедленной процедуры проведения измерений. Для того, чтобы откорректировать форму изображения, доступно несколько инструментов под кнопкой 9:

- **Поворот изображения.** Доступен поворот входного изображения на угол, кратный 90° : 90° по часовой стрелке, 180° , 90° против часовой стрелки. Угол аккумулируется, т.е. может быть совершено несколько последовательных операций поворота.
- **Отражение изображения.** При необходимости, может быть выполнено зеркальное отражение изображения как по вертикали, так и по горизонтали.

Также имеется возможность сбросить все примененные трансформации одной командой.

Кроме того, для выделения каких-либо деталей или сглаживания шумов доступно изменение содержимого изображения (кнопка 10). Доступны такие инструменты обработки изображения, как *Инвертирование* интенсивности пикселей рентгенограммы, *Повышение резкости* разной степени, *Выделение границ* и эффект *Тиснения*.

Д.3 Расширенный функционал

К расширенному функционалу разработанного программного комплекса относится автоматическая локализация коленных суставов на рентгенограмме,

выделение контуров костей сутсава описанным в данной работе алгоритмом и автоматическая классификация степени остеоартрита по шкале Келлгрена-Лоуренса для локализованных суставов. Данный функционал является опциональным и может быть использован по мере необходимости или отключен в случае отсутствия таковой.

Д.3.1 Особенности работы с модулем локализации

Модуль локализации SSD был написан на языке Python при помощи библиотеки машинного обучения **TensorFlow**. Данная библиотека имеет официальную поддержку API для языка C, которая и используется в данном проекте для целей детектирования коленного сустава. Для представленных нейросетевых моделей поддерживается как графический ускоритель, так и центральный процессор.

Для удобства был реализован собственный модуль **GraphProtoDetector**, позволяющий загружать модель, полученную при помощи обучения на языке *Python* и сохраненную в файл, в окружение пользовательского приложения и выполнять процесс обработки входного изображения, позволяя получить на выходе корректный набор ограничивающих прямоугольников, означающих обнаруженный на входном изображении коленный сустав. Данный модуль инкапсулирует в себя кодирование входного изображения в нужный для использования в детекторе SSD формат и декодирование результирующего вектора этого детектора в набор координат ограничивающих прямоугольников найденных на рентгенограмме суставов, оценки точности локализации и т.п.

Д.3.2 Выделение контуров

Выделение контуров сустава запускается для каждого изображения вручную при необходимости. Пользователь имеет возможность настроить детали работы алгоритма и может редактировать финальные контуры для более точного позиционирования — каждый найденный контур представлен графическим эле-

ментом **Умная кривая** и обладает всеми свойствами графических элементов, включая информацию о площади, периметре, средней яркости области внутри контура и т.п.

Д.3.3 Особенности работы с модулями классификации

Для обучений сверточных сетей использовался фреймворк **Pytorch**, написанный на языке *Python*. Основное пользовательское приложение при этом реализовано на языке C++. Для повторного использования полученных наработок при реализации применяется инструмент **TorchScript**, который позволяет использовать код, написанный при помощи Pytorch, в языке C++.

TorchScript представляет собой автономный компилятор языка Python, а также вспомогательные инструменты конвертации код из одного языка в другой, методы загрузки моделей, полученных при обучении на языке Python и др. Существуют как версии, использующие графический ускоритель, так и версии, работающие исключительно на центральном процессоре, которая и использовалась при реализации программного комплекса в рамках исследования.

Конвертация сверточной модели-классификатора из фреймворка Pytorch в формат, пригодный для использования в C++ при помощи TorchScript, осуществляется путем проведения трассировки графа вычислений для соответствующего классификатора и точного его воспроизведения в целевой программе, написанной на C++. Этот метод имеет определенные ограничения и не позволяет работать со сторонними библиотеками из-за отсутствия возможности проведения трассировки внутри методов этих библиотек, однако имеющегося функционала достаточно для точной эмуляции работы имеющихся классификаторов.

Д.4 Общее описание ПО

Иллюстрация интерфейса разработанного приложения представлена на рис. 79. Список загруженных изображений находится в нижней части интер-

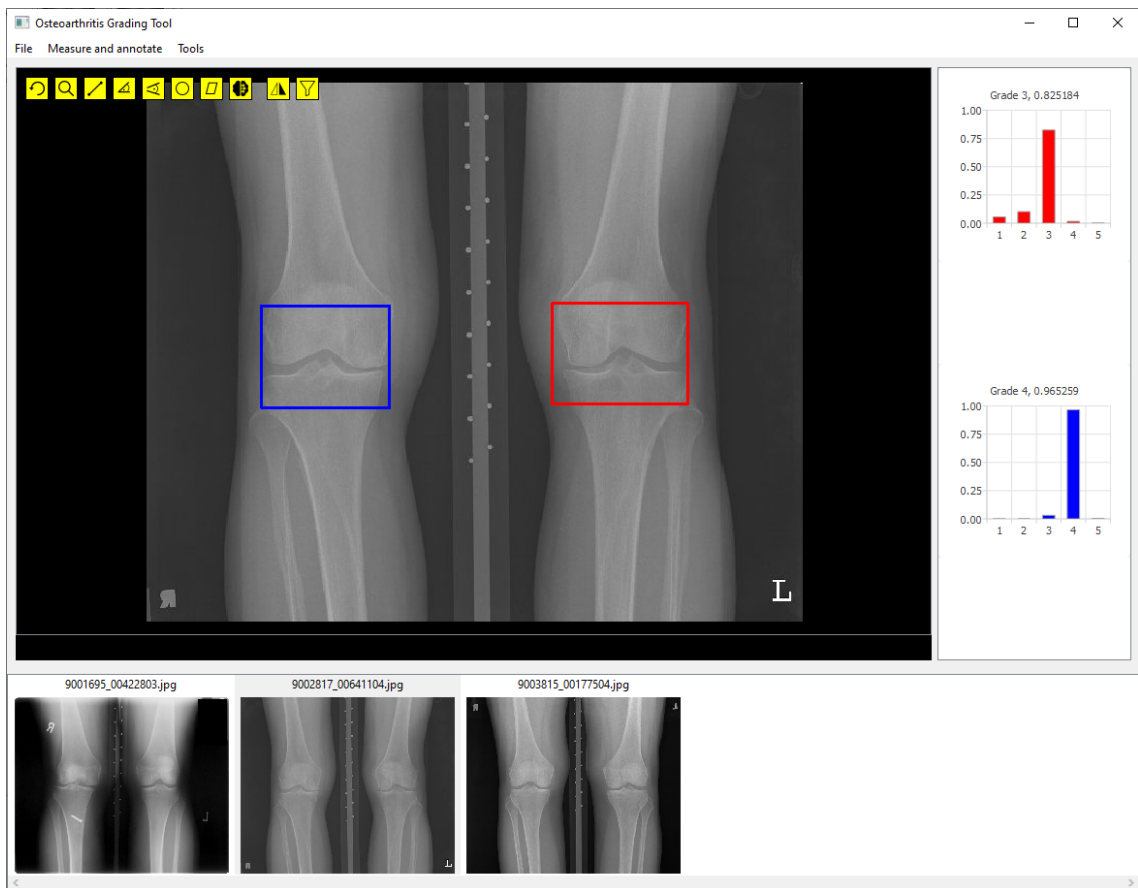


Рисунок 79 — Иллюстрация основного интерфейса программного обеспечения: вид после автоматической классификации остеоартрита

фейса, результат автоматической классификации остеоартрита в виде графиков распределения вероятностей того или иного класса по каждому коленному суставу — в правой части.

Инициализация модулей локализации и классификации выполняется при первой загрузке изображения и запуске процедуры автоматической классификации. В случае, когда автоматическая классификация не используется и отключена в настройках (т.е. пользователь пользуется набором доступных инструментов измерения вместо нее), этот подход позволяет не выполнять лишние операции и ускорить загрузку основного функционала приложения при запуске.