

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное бюджетное образовательное учреждение высшего образования  
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий

(наименование института полностью)

Кафедра «Прикладная математика и информатика»

(наименование)

01.04.02 Прикладная математика и информатика

(код и наименование направления подготовки)

Математическое моделирование

(направленность (профиль))

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
(МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ)**

на тему «Применение сверточной нейронной сети для дифференциации  
туберкулеза и пневмонии на цифровых рентгенограммах»

Студент

А.А. Спиркина

(И.О. Фамилия)

(личная подпись)

Научный

к.т.н., доцент, А.В. Очеповский

руководитель

(ученая степень, звание, И.О. Фамилия)

Тольятти 2020

## Оглавление

Введение.....	4
Глава 1 Анализ проблемы классификации патологий на рентгенограммах.....	8
1.1 Виды патологий на рентгенографических снимках .....	8
1.2 Обзор современного состояния классификации рентгенографических изображений .....	13
1.3 Анализ программных средств для решения проблемы классификации .....	19
1.3.1 Анализ языков программирования .....	19
1.3.2 Анализ формата DICOM .....	22
1.3.3 Исходные данные задачи классификации .....	24
Глава 2 Формирование решения задачи классификации рентгенографического снимка грудной клетки .....	28
2.1 Методы сегментаций и обработки изображения .....	28
2.2 Архитектура и методы обучения сверточной нейронной сети .....	35
2.3 Анализ моделей, предварительно обученных глубоких сверточных нейронных сетей .....	37
2.4 Структура математической модели сверточной нейронной сети.....	41
Глава 3 Формирование составной глубокой сверточной нейронной сети для задачи классификации рентгенографического снимка .....	47
3.1 Обработка рентгенографических снимков.....	47
3.1.1 Обработка формата DICOM .....	47
3.1.2 Сегментация рентгенографического снимка .....	49
3.2 Подготовка выборки для нейронной сети .....	52
3.3 Реализация сверточной нейронной сети.....	53
3.3.1 Реализация сверточной нейронной сети на основе модели Inception-v3 .....	54
3.3.2 Реализация сверточной нейронной сети на основе модели VGG-1660 .....	54
Глава 4 Проведение эксперимента на эффективность работоспособности алгоритма .....	65

4.1 Сравнительный анализ результатов классификаций .....	65
4.2 Классификация рентгенографических снимков глубокой сверточной нейронной сетью .....	70
Заключение .....	73
Список используемых источников.....	75
Приложение А_Код составной глубокой сверточной нейронной сети на основе модели Inception-v3 .....	79
Приложение Б_Код составной глубокой сверточной нейронной сети на основе модели VGG-16 .....	81

## Введение

Острая респираторная вирусная инфекция является группой клинически и морфологически подобных острых воспалительных заболеваний органов дыхания.

По данным в 2016 году зафиксировано 297 миллионов случаев заболеваний легких по всему миру, из которых 2,2 миллиона со смертным исходом. По данным Росстата в Российской Федерации за этот же период легочными заболеваниями переболело более 971 тысяч человек, умерло более 56 тысяч. В 2019 году наблюдался скачок заражения людей вирусной пневмонией, что привело к высокой смертности, которая составила 19 тысяч в первое полугодие [26].

Для того, чтобы предотвратить развитие и осложнение респираторных заболеваний, используется диагностика легких.

Рентгенография грудной клетки является наиболее часто используемым диагностическим методом для исследований заболеваний легких, таких как рак, отек легких, плевральный выпот, пневмония, туберкулез.

Многие лекарства эффективны для большинства заболеваний только на ранней стадии [8]. Экранирование может помочь ранней диагностике, но для обеспечения массового применения необходимо использовать метод более чувствительный и экономичный. Стандартная рентгенография грудной клетки отвечает этим требованиям, за исключением того, что современные методы имеют небольшую неточность. Более удобный метод, имеющий высокую точность, это компьютерная томография (КТ), но данный метод имеет недостаток – КТ подвергают пациента более высокой дозе радиации. Еще одной причиной широко распространенного использования обычной рентгенографии является его экономическая целесообразность [11].

Несмотря на целесообразность использования рентгенографии, на стадии постановки диагноза врачом существует вероятность того, что болезнь может

быть не обнаружена. В связи с этим появляется риск развития патологии, что может привести к летальному исходу.

На сегодняшний день существуют компьютерные технологии, которые облегчают нахождение патологии и повышают точности постановки диагноза. Нахождение и распознавание патологий на рентгенографических снимках помогают решить методы обработки изображения и нейронная сеть, которая, основываясь на большом количестве заведомо верных результатов постановки диагноза, выявляет патологии на снимках с минимальной погрешностью. Для того, чтобы нейронная сеть была способна выполнять поставленную задачу, требуется обучить её алгоритму обнаружения области заболевания. Также для облегчения решения данной задачи существуют методы сегментации с различными вариациями точности результата.

Актуальность темы исследования обусловлена проблемой обнаружения и классификации патологий на рентгенограммах грудной клетки.

Целью диссертационной работы является уменьшение коэффициента ошибки при постановке диагноза с помощью классификации патологий на рентгенографических снимках нейронной сети.

Объектом исследования выступает процесс классификации патологий на рентгенографическом снимке грудной клетки.

Предметом исследования составила глубокая сверточная нейронная сеть для классификации патологий на рентгенографическом снимке грудной клетки.

Гипотеза исследования: разработанная нейронная сеть послужит вспомогательным инструментом для врача в диагностике заболеваний, что уменьшит коэффициент неточно поставленных диагнозов и сократит время на обследования.

Задачами в диссертационной работе являются:

1. Проанализировать диагностику легочных заболеваний и их проявление на рентгенографических снимках.
2. Проанализировать современное состояние методов классификации рентгенографического снимка.

2. Сформулировать задачи классификации патологий на рентгенографических снимках.

3. Провести сравнительный анализ и выбор подходов к классификации изображений.

4. Разработать систему для классификации патологий легких на рентгенограммах.

5. Произвести тестирование эффективности реализованной системы.

Теоретической основой исследования стали научные достижения отечественных и зарубежных ученых, занимающиеся диагностикой заболеваний при помощи искусственного интеллекта. Авторы указывают, что для достижения высокой точности результата классификации используются модели заранее обученных нейронных сетей. Упор в распознавании делается на объеме и форме набора данных, зачастую используются необработанные изображения, тем самым, полученный результат, является недостаточно высоким.

Практическая значимость диссертационного исследования заключается в применении системы в качестве вспомогательного инструмента врача в диагностике заболеваний, что позволит повысить эффективность и уменьшить коэффициент ошибки.

Методы исследования, которые использовались в процессе формирования диссертационной работы: анализ и синтез модели, математическое моделирование, экспериментальные измерения и анализ.

Диссертационное исследование производилось с 2018 по 2020 гг. в три этапа:

1. Констатирующий этап исследования (2018 г.) состоял в формализации темы, цели, задач, гипотезы исследования, подтверждении актуальности решения проблемы, произведения обзора современного состояния темы и определении методики решения задач.

2. Моделирующий этап (2018-2019 гг.) состоял в обзоре и выборе методов классификаций, моделировании математической модели, апробации результатов исследования на научных конференциях и формализации статей.

3. Экспериментальный этап (2020 г.) состоял в реализации метода обработки рентгенографического снимка для последующей формализации набора данных, реализации глубокой сверточной нейронной сети, анализе полученных результатов работоспособности системы.

Публикации по теме магистерской работы представлены на XLV Самарской областной студенческой научной конференции, VI международной научно-практической конференции. Результаты исследования сформулированы в статьях в научных сборниках.

Научная новизна заключается в получении высокой точности классификации пневмонии и туберкулеза на рентгенографических снимках грудной клетки при помощи моделировании составной глубокой сверточной нейронной сети, тонкой настройки и составлении набора данных, помогающих повысить точность.

На защиту предоставляются:

1. Метод предобработки рентгенографических снимков.
2. Математическая модель сверточной нейронной сети.
3. Реализованная глубокая сверточная нейронная сеть.

Диссертационная работа состоит из введения, четырех глав, заключения, списка литературы и трех приложений. Работа изложена на 74 страницах, включает 35 иллюстрации, 10 таблиц, 35 использованных источников.

# **Глава 1 Анализ проблемы классификации патологий на рентгенограммах**

## **1.1 Виды патологий на рентгенографических снимках**

Понимание того, как выглядит и где возможно расположение той или иной патологии заболевания легких на рентгенографическом снимке грудной клетки является важным аспектом в решении проблемы классификации.

Для исследования заболевания органов дыхания на первом этапе применяются нативные рентгенологические методики. Самые часто применяемые из них – это флюорография, рентгенография и линейная томография. Выводом при применении первых двух методов являются снимки с изображением теневого рисунка легких. При томографии может быть получен определенный срез органа в нужной толщине.

Для исследования были выбраны рентгенографические снимки грудной клетки в прямой проекции, так как это изображение дает возможность полно оценить вид легочного рисунка и прилегающих к нему органов, также данный вид снимков наиболее часто используют врачи в обследовании.

Нормальный легочный рисунок представляет собой общее теневое изображение грудной клетки (рисунок 1). При прямой проекции рисунок складывается из светлых полей, симметрично расположенных в боковых отделах грудной полости (легкие), и находящейся между ними срединной тени. Снизу грудная полость отделена от полости живота диафрагмой. Снаружи по бокам видна тень грудной стенки [17].



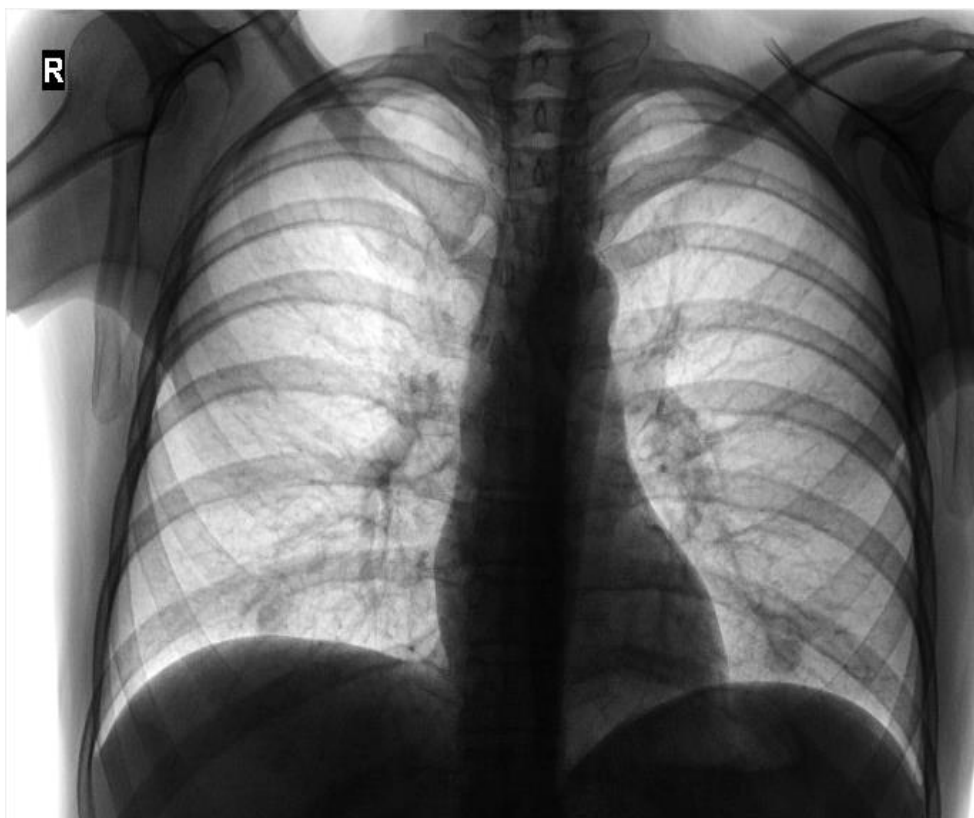


Рисунок 1 – Легочный рисунок

Легочные поля пересекаются полосовидными тенями ребер. Их задние отделы идут от позвоночника, расположены горизонтально, выпуклостью обращены вверх, имеют меньшую ширину и большую интенсивность тени. Передние отделы ребер идут от грудной стенки косо сверху вниз, выпуклостью обращены вниз, их тень менее интенсивная и более широкая. Их концы, образованные хрящевой тканью, которая не поглощает рентгеновские лучи, как бы обрываются примерно на уровне срединно-ключичной линии.

Срединную тень в прямой проекции образуют в основном сердце, аорта и позвоночник. Из частей грудины в этой проекции видна только ее рукоятка с грудино-ключичным сочленением [15].

Патологические проявления на снимке имеют разнообразный вид, но различают четыре основных группы изменения рисунка:

- затенение легочных полей;
- просветление легочных полей;
- изменение легочного рисунка;

- изменение корней легких.

Данные группы детализированы по протяженности, форме, структуре и очертанию. По ним выделяются девять рентгенологических синдромов, которые могут отразить практически все разнообразные отображения паталогических заболеваний (рисунок 2).

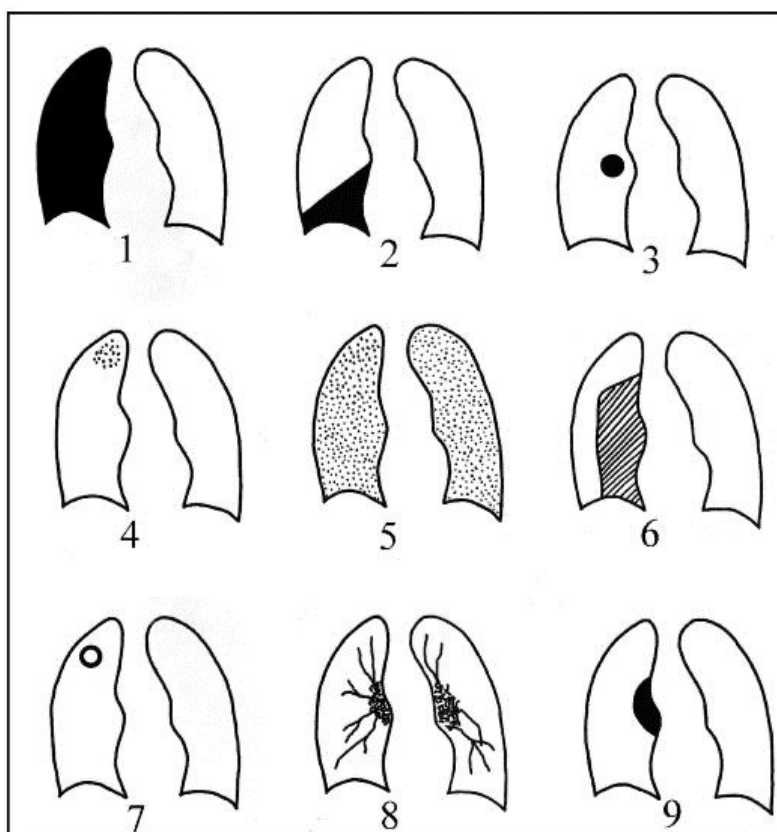


Рисунок 2 – Рентгенологические синдромы заболеваний, обширное затемнение(1), ограниченное затемнение (2), круглая тень (3), очаги (4), обширная очаговая диссеминация (5), обширное просветление (6), ограниченное просветление (7), изменение корней легких (8), изменение легочного рисунка(9)

При этом каждый из этих синдромов может отображать несколько заболеваний в зависимости от вида и стадии развития. Многие заболевания могут иметь схожее формирование, и заключить, какое оно именно, является трудной задачей. В этом заключается проблема распознавания той или иной патологии. Также на само нахождение очага заболевания иногда требуется

достаточно много времени, так как оно может быть выражено не явно и в зависимости от множества факторов быть скрыто.

По положению затенения возможно определить к какому патологическому процессу она принадлежит, исключая жидкость в междолевом расположении (таблица 1).

Таблица 1 – Характер затенения при патологическом заболевании

Положение средостения	Однородное затенение	Неоднородное затенение
Не смещено		Воспалительная инфильтрация
		Отек легкого
Смещено в сторону затенения	Ателектаз	Плевральные шварты
	Отсутствие легкого	Цирроз легкого
Смещено в противоположную сторону	Жидкость в плевральной полости	Большое новообразование
	Большое новообразование	

Наиболее часто встречающимися заболеваниями являются: пневмония; бронхит; туберкулез; вода в легких; инфильтрация; пневмоторакс.

Данные заболевания имеют определенное выражение на рентгенографическом снимке. Пневмония, туберкулез подробно рассмотрены на примерах снимков грудной клетки, так как за период 2018-2019 гг. этот диагноз по данным Росстата был поставлен более 140 тысяч человек [26]. Поэтому эти заболевания наиболее целесообразны для анализа.

Пневмония – воспаление легочной ткани инфекционно-воспалительного характера, в который вовлекаются все структурные элементы легочной ткани [15].

На рисунке 3 представлен снимок грудной клетки, где в правой доле в верхней части имеется ограниченное затемнение пневмонии.

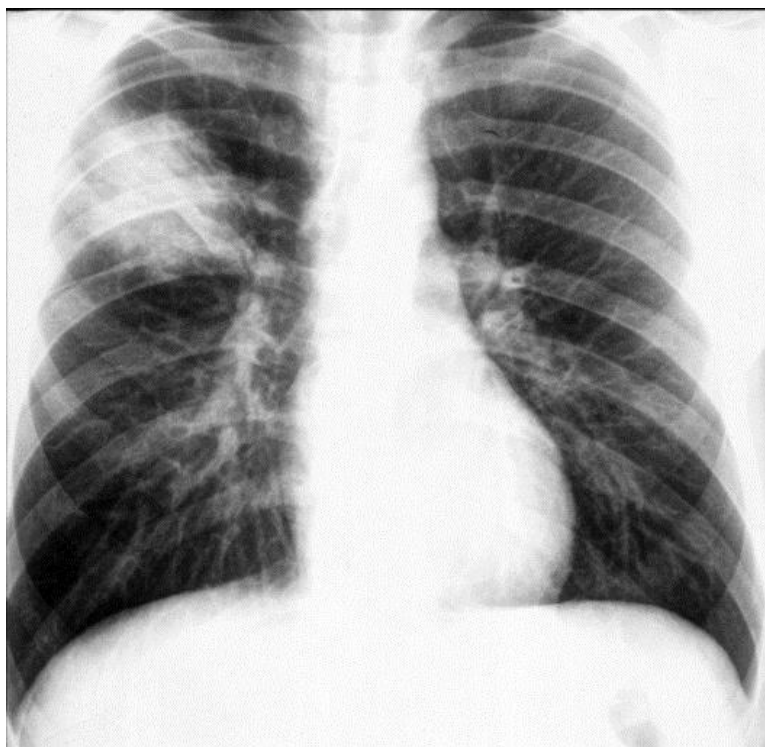


Рисунок 3 – Пневмония

В зависимости от вида пневмония по-разному проявляется на снимке: очаговые затемнения с неровными контурами; небольшие тени (1-1,5 см) со слабой или умеренной интенсивностью, неоднородной структурой и нечеткими границами, нарушение рисунка корней легких; жидкость в полостях плевры, инфильтрация одной из долей, расширение корней; уплотнение корней легочного рисунка; обширные затемнения, утолщение плевры, наличие полостей, наполненных жидкостью; треугольные пятна с однородной структурой, светлые очаги, приподнятая диафрагма [17].

Туберкулез – инфекционное заболевание, вызванное микобактериями туберкулеза [15].

На рисунке 4 изображена одна из форм туберкулеза – диссеминирование легких.

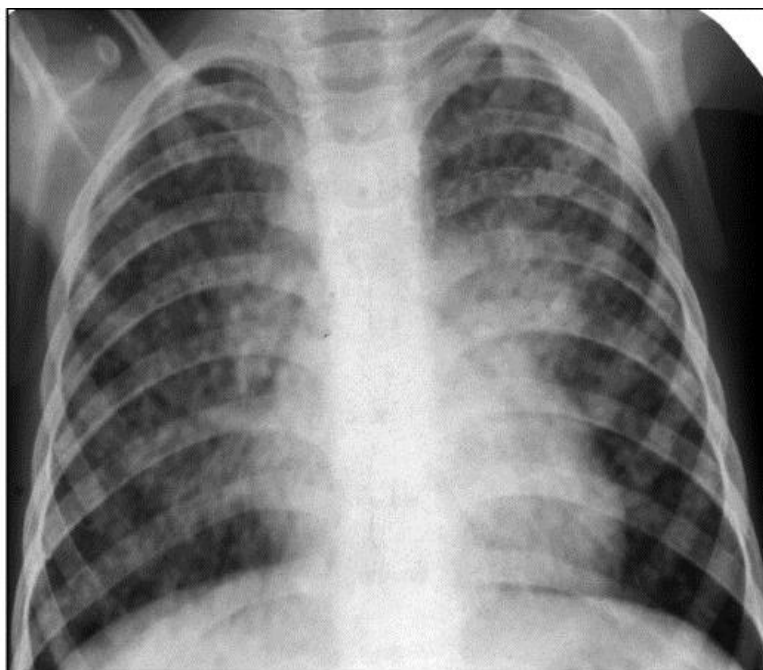


Рисунок 4 - Туберкулез

Туберкулез может быть выражен: расширением корней; ограниченным затемнением, выраженным круглой тенью; множественными очагами, сформированными в локализующихся пределах; поражением одной или двух долей «высыпанием» множественными разновидностями теневых очагов; ограниченным просветлением [17].

Инфекционные заболевания пневмония и туберкулез наиболее распространены и их лечение необходимо проводить своевременно, на начальной стадии. При запуске патологий могут появиться серьезные последствия, вплоть до летального исхода. Поэтому данные заболевания наиболее целесообразнее использовать для дальнейшего исследования.

## **1.2 Обзор современного состояния классификации рентгенографических изображений**

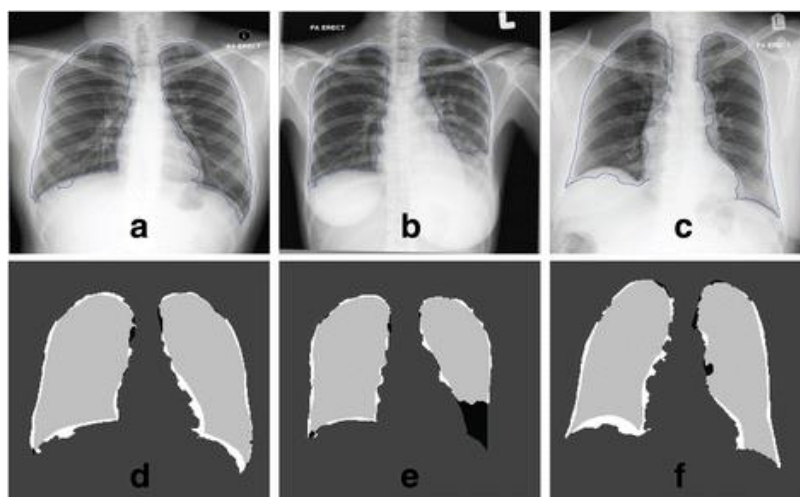
Существует множество способов для обнаружения образований на рентгенографическом снимке грудной клетки. Методы можно подразделить на

две группы: методы сегментации и обнаружения узелков заболевания, не требующие обучения; методы распознавания с использованием нейронной сети.

С помощью методов сегментации возможно облегчить нахождение патологий. Например, выделение легочной структуры и отсечение не нужных для анализа областей снимка.

В работе «Сегментация легких на стандартных и мобильных рентгенограммах грудной клетки с использованием ориентированного фильтра Гаусса» метод основан на ориентированном фильтре Гаусса в сочетании с кластеризацией k-means. Алгоритм является полностью автоматизированным. Он состоит из двух этапов: предварительная обработка и этапа сегментации. На первом этапе снимки, полученные на разных платформах, в разных положениях человека, обрабатываются. Производится выравнивание грудной клетки и корректировка цвета путем растяжения гистограммы.

На втором этапе автоматизированный метод адаптировал ориентированный гауссовый фильтр для нахождения пороговых значений, чтобы можно было определить контур легких. Затем он заполняется с помощью глобального порогового значения, а конечный результат уточняется с помощью нескольких различных кластеров по методу k-means. В результате работы алгоритма получен сегментированный легочный рисунок с точностью в 77%. При фильтрации, которая засвечивает доли, рисунок может быть восстановлен с помощью k-means (рисунок 5) [35].



## Рисунок 5 – Результат работы алгоритма

Также возможно удаление с легочного рисунка ребер и ключиц. Данный метод полезен при нахождении патологий рака, так как тени костей ребер могут перекрывать очаги. В работе «Автоматический метод сегментации легких и удаление ребер для лучшего обнаружения узелков на рентгенограммах грудной клетки» представлено программное обеспечение, которое подавляет видимость ребер и ключиц. Приложение через алгоритм распознавания обнаруживает области, соответствующие костям, и через обработку изображения создает акцент на ней. Найденная область вычитается из начального изображения, создавая акцент на мягких тканях. После подавления ребер усиливается контраст мягких тканей легких [28].

Таким образом, возможно обнаружить тени узелков рака на ранней стадии или его начальные признаки. Чаще всего программное обеспечение обнаруживало патологии за скрытой тенью костей. Данный метод был протестирован и в эксперименте участвовали врачи с разным уровнем опыта. Алгоритм повышает точность нахождения узелков рака.

В схожей по тематике работе «Улучшенное обнаружение патологий с помощью программного обеспечения, которое затемняет ребро и ключицу на рентгенограммах грудной клетки» «удаление» костной структуры происходит по средствам использования фильтра Габора. Этот метод более устойчив к шумам и не требует большой затраты времени на определение энергетических зависимостей костной ткани от мягких тканей [32].

Представленные методы сегментации и обработки рентгенографических снимков дают возможность повысить результативность, точность поиска и определения патологии, но большую часть работы по распознаванию выпадает на долю врача. Также данные алгоритмы ограничены в распознавании различных видов патологий. При этом во второй разработке требуется определенный вид рентгенографических снимков, что может затруднить работу.

Распознавание различных типов патологий возможно с помощью нейронной сети. Обычно в таких целях применяются сверточные нейронные сети. Их архитектура основана на архитектуре неокогнитрона, который упрощен и дополнен обучением алгоритма обратного распространения ошибки. Неокогнитрон основан на строении зрительной системы, он способен распознавать образы независимо от их положения, масштаба, искажения. Таким образом, сверточная нейронная сеть устойчива к изменению и искажению объекта на изображении, что означает, что нет необходимости подстраивать рентгенографические снимки для распознавания.

Сверточная нейронная сеть состоит из многих слоёв. Слои бывают двух типов: сверточные (Convolutional) и подвыборочные (Subsampling), сверточные и подвыборочные слои чередуются друг с другом. В каждом слое имеется набор из нескольких плоскостей, причём нейроны одной плоскости имеют одинаковые веса, ведущие ко всем локальным участкам предыдущего слоя (как в зрительной коре человека), изображение предыдущего слоя как бы сканируется небольшим окном и пропускается сквозь набор весов, а результат отображается на соответствующий нейрон текущего слоя. Таким образом набор плоскостей представляет собой карты характеристик (feature maps) и каждая плоскость находит «свои» участки изображения в любом месте предыдущего слоя. Следующий за свёрточным слоем подвыборочный слой уменьшает масштаб плоскостей путём локального усреднения значений выходов нейронов. Таким образом, достигается иерархическая организация. Последующие слои извлекают более общие характеристики, меньше зависящие от искажений изображения.

В работе «Автоматизированная классификация типов рака по цитологическим изображениям с использованием глубоких сверточных нейронных сетей» была представлена технология в качестве «второго мнения» для поддержки диагностики патологии и оценки злокачественных новообразований. Глубокое обучение нейронной сети используется для получения лучшей производительности и точности результата. Нейронная сеть



была обучена распознавать три вида раковых образований. Нейронная сеть состоит из трех сверточных слоев, трех объединенных слоев и двух полностью соединенных слоев [34]. В результате нейронная сеть распознает очаги заболеваний с 70% точностью.

Классификацию снимков грудной клетки рассматривают в работе «Высокопроизводительная классификация рентгенограмм с использованием глубоких сверточных нейронных сетей». Предоставленные для исследования снимки в формате DICOM были преобразованы в формат JPEG для снижения размера и удаления, не имеющих значимости метаданных. Так же для увеличения наборов для обучения и тестирования с помощью преобразований за счет ориентирования снимков. Таким образом, набор данных был увеличен в 3 раза.

За основу нейронной сети была выбрана уже предварительно обученная модель сверточной нейронной сети GoogLeNet. Она содержит 22 сверточных слоя, включая 9 начальных модулей. Данная модель была предварительно обучена на более миллионах изображениях.

Некоторые слои были переобучены для задачи распознавания рентгенограмм в 30 эпох.

Точность результата работы данной сверточной сети составляет 95%. Таким образом, в статье делается вывод, что предварительно обработанные изображения, увеличенный набор данных влияет на точность результата. Уже обученная сверточная нейронная сеть показывает высокий процент точности, а также уменьшает время, затраченное на обучение сети [29].

В работе «Глубокая сверточная нейронная сеть для выявления заболеваний органов грудной клетки» анализируются работоспособность некоторых нейронных сетей в классификации рентгенографических снимков. Рассматривается архитектура нейронной сети обратного распространения ошибки, конкурентной нейронной сети и сверточной сети. Данные сети были обучены на более 100 тысячах снимках из 5 классов заболеваний. Размер

входных данных были снижены до 32x32 пикселей в связи с уменьшением нагрузки при обучении.

Использовались модели BPPN1 – BPPN4 нейронной сети обратного распространения ошибки, модели CrNN1 – CrNN3 конкурентной сети и CNN сверточная нейронная сеть. Лучший результат распознавания показала сверточная нейронная сеть, достигнув 90% точности. Высокая точность распознавания сверточной нейронной сети связана с тем, что другие сети не извлекают важные признаки из входных изображений.

Так же в данной статье анализируется работоспособность моделей сверточных нейронных сетей уже предварительно обученных на больших базах данных изображений. Были продемонстрированы точности распознаваний моделей VGG16, точность которой составляет 86%, и VGG19 – 93% точности распознавания.

На основе полученных результатов в статье заключается вывод, что для задачи классификации рентгенографических снимков грудной клетки целесообразнее использовать модели сверточных нейронных сетей, которые уже были предварительно обучены на обширных базах данных, чем строить модель нейронной сети «с нуля» [27].

Таким образом, самым эффективным методом для нахождения и распознавания очагов заболевания является сверточная нейронная сеть. Она устойчива к искажению снимка, так как не всегда грудная клетка может находиться в правильном положении, обладает высокой точностью результата распознавания и быстротой обучаемости. Нейронную сеть возможно обучить распознавать несколько различных видов патологий. На точность результата влияет объем и вид наборов данных. Для классификации рентгенограмм целесообразнее использовать уже предварительно обученную сверточную нейронную сеть, которую необходимо переобучить под задачу исследования. Также прослеживается влияние формы набора данных на точность результата: где имеется предобработка в виде сегментирования необходимых для анализа

областей, точность результата выше, чем у сетей, обученных на «сырых» снимках.

### **1.3 Анализ программных средств для решения проблемы классификации**

#### **1.3.1 Анализ языков программирования**

На данный момент существует множество программных средств, которые дают возможность работать с изображениями и нейронными сетями.

Высокоуровневый язык программирования MATLAB используется во всех аспектах вычислительной математики. Обширная встроенная библиотека функций, которая облегчает работу во время написания кода. В основном используется матричное представление данных [18].

MATLAB дает возможность обрабатывать изображения. Встроенные функции фильтров, а также матричное представление данных уменьшают время решения задач и упрощает процесс. Для построения функций возможно вычисление статистических свойства объектов.

Язык поддерживает функцию загрузки и работы с форматом DICOM, что является достоинством этого языка в данном исследовании.

С помощью Deep Learning Toolbox предоставляется функции командной строки и приложения для создания, обучения и моделирования неглубоких нейронных сетей. Приложения позволяют легко разрабатывать нейронные сети для таких задач, как классификация, регрессия (включая регрессию временных рядов) и кластеризация. После создания сетей в этих инструментах можно автоматически создавать код MATLAB для записи своей работы и автоматизации задач [18].

Данный язык прост в понимании и формировании кода, но для вычисления объемных данных он требует высокой вычислительной мощности, что может затруднить процесс вычисления. Также недостатком данного языка является сложная организация в написании глубоких нейронных сетей.

Универсальный, объектно-ориентированный язык программирования - Java. Данный язык строго типизирован. Позволяет формировать стандартные программы и повторно используемый код. Язык многопоточен и возможно выполнение множества задач параллельно. Имеет разнообразные библиотеки для работы с графикой и звуком, что позволяет легко написать любое приложение. Производителен и имеет Garbage collection, которое автоматически управляет памятью [21].

Для сегментации изображений использует готовые инструменты с открытым кодом IMMI и библиотека JAI. Данное API позволяет создавать собственные процедуры обработки изображений.

Приложение с открытым кодом dcm4che позволяет работать с форматом DICOM на Java: чтение, извлечение из архива, запись изображения. Работа с форматом в основном заключается в просмотре самого снимка.

Библиотека для реализации нейронных сетей Weka в основном используется для обработки наборов данных. MOA (Massive Online Analysis) предназначена для потоковой обработки набора данных в режиме реального времени. MALLET инструментальное средство машинного обучения для изучения текстов.

Python – высокоуровневый язык программирования, поддерживающий структурированную и объектно-ориентированную парадигму программирования [19].

Существует множество библиотек, которые поддерживают разнообразный функционал во многих сферах применения, такие как, обработка изображений, текстов, создание приложений, технические вычисления, работа с базами данных и так далее.

Python предоставляет библиотеки, с помощью которых возможно работать и обрабатывать изображения: scikit-image, SciPy, NumPy, PIL/Pillow, OpenCV-Python, SimpleITK, SimpleCV. Пакет NumPy один из основных библиотек, который поддерживает работу с массивами. Подгружаемое изображение представляется в виде матрицы, становясь доступным для

операций. В коллаборации с этим пакетом работают многие другие библиотеки обработки изображений. Также NumPy применяется в машинном обучении, высококачественные библиотеки используют его для манипулирования Tensors.

Данный язык предоставляет возможность легкой работы с нейронными сетями за счет библиотеки Keras. Keras является надстройкой над фреймворками TensorFlow и Theano. Возможна работа на процессоре, так и на GPU, что позволяет распараллелить процессы и уменьшить время, затрачиваемое на обучение. Библиотека SciPy содержит модули для работы со статистикой, математическими функциями, а также с обработкой изображений. Pandas является библиотекой для подготовки наборов данных для обучения. С помощью нее осуществим легкий анализ данных, их извлечение и структурирование [25].

TensorFlow – открытая библиотека для машинного обучения. Она позволяет решать задачи формирования, настройки и обучения глубоких нейронных сетей. Библиотека квалифицируется на нейронных сетях, которые классифицируют и находят образы на изображениях. API доступен для работы с Python, C++, Java.

Библиотека является кроссплатформенной. Она работает практически на всех графических процессорах, включая мобильные и встроенные платформы, также на тензорных платформах TPU [23].

Рассмотренные языки программирования были проанализированы по поддерживаемому функционалу, который необходим для выполнения цели и задач исследования. Результат анализа был сведен в таблицу 2.

Таблица 2 – Сравнительный анализ языков программирования

Необходимый функционал / Программное средство	MATLAB	Java	Python
Работа с	Да	Да	Да

изображениями			
Сегментация изображений	Да	Да	Да
Поддержка работы с форматом DICOM	Да	Да	Да
Возможность обработки формата DICOM	Да	Нет	Да
Работы с глубокими нейронными сетями	Да	Да	Да
Простая работа с готовыми моделями сверточных нейронных сетей	Нет	Да	Да

Основываясь по данным таблицы, можно сделать вывод, что для исследования наиболее подходящими языком программирования является Python. Данный язык предоставляет простую работу с глубокими сверточными нейронными сетями, обработкой изображения, форматом DICOM.

### **1.3.2 Анализ формата DICOM**

Цифровая визуализация и коммуникации в медицине (DICOM) является стандартом для передачи и управления медицинской информации, визуализации связанных с ними данных [16]. DICOM наиболее часто используется для хранения и передачи медицинских изображений.

DICOM группирует информацию в наборы данных. Например, файл рентгеновского снимка грудной клетки может содержать идентификатор пациента, по которому изображение никогда не может быть ошибочно идентифицировано. Объект данных DICOM состоит из ряда атрибутов, включая такие элементы, как имя, идентификатор, серия кадров снимка органа (рисунок 6).

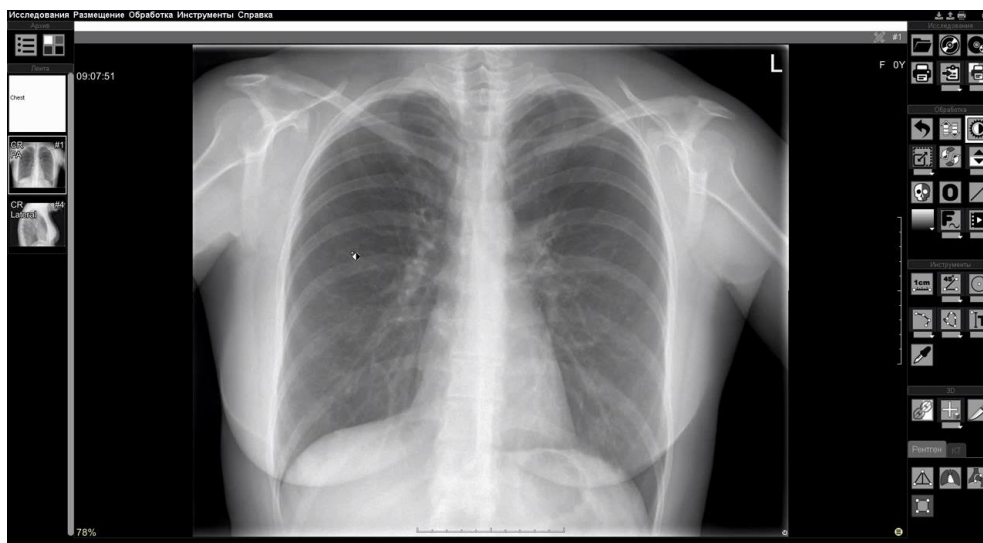


Рисунок 6 – Рентгенографический снимок в формате DICOM

Для просмотра рентгенографических снимков используется шкала Хаунсфилда. Она произвольно присваивает воде плотность 0 единиц Хаунсфилда (HU) и воздуху значение -1000 HU. Плотные материалы, такие как кость, имеют значения плотности, приближающиеся к +1000 HU. Плотность Хаунсфилда тканей отражает их ослабление рентгеновского излучения и пропорциональна их физической плотности.

Цветовая гамма серых оттенков назначается тканям различным оттенком с помощью КТ, изменяясь в зависимости от плотности. Цветовая шкала HU – это шкала серого оттенка, от черного до белого. Менее плотные ткани кажутся более черными, более плотные ткани – более белыми, а ткани с промежуточной плотностью занимают промежуточные серые оттенки. Поскольку человеческий глаз и мозг не могут воспринимать тонкие вариации серых оттенков, обычно

нецелесообразно равномерно распределять доступные серые оттенки по всему диапазону плотностей при отображении изображения. Вместо этого распределение серых оттенков варьируется в зависимости от ткани наибольшего интереса, что устанавливается с помощью шкалы (таблица 3) [31].

Таблица 3 – Плотность тканей по шкале Хаунсфилда

Ткань	Центральное значение HU	Диапазон HU
Легкое	-498	+1465
Кость	+570	+3077
Мягкая ткань	+56	+342
Кровь	+40	+200
Воздух	-354	+856

Центральное значение – это плотность Хаунсфилда, которому присвоен центральный серый оттенок. В зависимости от него назначаются крайние оттенки белого и черного: через сложение белый, через вычитание черный.

С помощью денситометрических показателей вычисляются положения и область интересующего объекта, концентрируется внимание на тканях, где наиболее четко отображается патология. Так же с одного снимка возможно получить несколько изображений с разными плотностями, для дифференциации различных патологий.

### **1.3.3 Исходные данные задачи классификации**

Методы обработки и классификации изображений с помощью искусственного интеллекта являются распространенной задачей в рентгенологии. Благодаря этой технологии возможно решить множество сложных проблем, которые были трудны для людей. Искусственный интеллект, выявляя значимые взаимосвязи в исходных данных, используется в поддержке диагностики, лечении и прогнозировании исходов во многих медицинских ситуациях.



Искусственный интеллект в медицине имеет важное значение, поскольку он потенциально может оптимизировать работу с огромным потоком данных. Обеспечение быстрой интерпретации и контекстуально значимой информации для врача позволит создать второе мнение. Также он повысит клиническую надежность – искусственный интеллект поможет врачам формировать более точное мнение, так как в огромном потоке данных может быть упущена важная информация.

Изначально для распознавания рентгенографических снимков использовались простые методы линейной регрессии и сравнительного прогона изображений, но точность результатов составляла низкий процент. Позже в диагностике стали применяться многослойные, вероятностные, регрессионные нейронные сети. В выявлении заболеваний легких использовался рентгенографический снимок, который предварительно обрабатывался: производилось выравнивание гистограммы изображения, сегментирование легочных долей; затем применялась нейронная сеть для классификации заболеваний. Данный метод эффективно используется для дифференциации, однако он значительно проигрывает в использовании глубокой сверточной нейронной сети. Глубокое обучение способствует повышению точности, снижению времени анализу данных и минимальной квадратной ошибки.

По обзору рентгенографических снимков легочного рисунка и отображения на них патологий, также по современному состоянию задачи классификации патологий на снимках, анализу программных средств и на основе цели диссертационной работы были сформулированы требования.

В составление набора данных для классификации должны использоваться рентгенографические снимки с пневмонией, туберкулезом, нормой и другими патологиями. Рентгенографические снимки в формате DICOM необходимо конвертировать в формат JPEG, который сжимает данные без потерь и поддерживает полутоновый формат изображения. Это позволит уменьшить размер изображения, сократить время обучения, так же расширить набор данных для обучения, тестирования, проверки нейронной сети. Поэтому

должна быть произведена фильтрация по шкале Хаунсфилда для послойного отображения грудной клетки в трех состояниях.

Подготовить несколько наборов данных для классификации. Произвести сегментирование с наиболее точным результатом, где легочная структура отделена от неинформативных областей и где произведено увеличение контраста для повышения точности снимка.

Классификацию заболеваний необходимо получить с наиболее точным процентом. Алгоритм в конечном результате должен показывать, к какому классу принадлежит патология на рентгенографическом снимке.

Для выполнения требований необходимо рассмотреть методы сегментации, реализовать и выбрать наиболее подходящие для задачи. Обработать снимки в формате DICOM и сформировать несколько наборов данных. Проанализировать архитектуры глубоких сверточных нейронных сетей, математически смоделировать нейронную сеть и реализовать нейронную сеть.

Данная технология поможет в проведении диагностики с целью повышения точности постановки верного диагноза, снижении ошибки и станет вспомогательным инструментом врача.

## Вывод к главе 1

1) В современных публикациях по теме классификации изображений наиболее высокие результаты показывают глубокие сверточные нейронные сети по сравнению с другими способами распознавания.

2) Наиболее эффективными являются предварительно обученные нейронные сети. Для переобучения они требуют объем данных и времени значительно меньше, чем при обучении заново созданной нейронной сети.

3) По анализу работ, связанных с классификацией рентгенографических снимков, предварительно обработанные снимки, которые состоят из информативных для задачи областей, влияют на точность результата сверточной нейронной сети.

4) По проанализированным программным средствам для реализации задачи исследования наиболее близко подходит Python. Данный язык отвечает всем критериям и предоставляет простое решение процесса реализации.

5) Наиболее тяжелыми и часто встречающимися заболеваниями органов дыхания являются туберкулез и пневмония, так как за период 2018-2019 гг. этот диагноз по данным Росстата был поставлен более 140 тысяч человек. Поэтому целесообразнее использовать именно эти заболевания для обучения нейронной сети.

## **Глава 2 Формирование решения задачи классификации рентгенографического снимка грудной клетки**

### **2.1 Методы сегментаций и обработки изображения**

В бакалаврской работе «Разработка и реализация метода сегментации рентгенограммы с наименьшим коэффициентом ошибки» реализован гибридный метод сегментации, состоящий из однопроводного метода и бинаризации с оптимальным пороговым значением. Данный метод позволяет выделить 88% площади легочных долей.

Предлагается улучшить данный алгоритм детализацией снимка, путем уточнения краев и повышения контраста сегментированных областей. Для этого рассматриваются способы сегментации и обработки изображения.

Сегментация – разделение изображения на отдельные области, содержащие пиксели с похожими атрибутами [20]. Содержательные и полезные для анализа и интерпретации в изображениях регионы должны иметь тесную связь с объектами, которые представляют интерес. Значимая сегментация – это первый шаг низкоуровневой обработки изображений, преобразование из полутонового или цветного изображения в одно или несколько других изображений для получения изображения высокого уровня. Успех анализа изображений зависит от надежности сегментации, но точное разбиение изображения является очень сложной задачей.

Чтобы получить полезную информацию из изображения, процесс сегментации разбивает изображение на неклассифицированные «примитивы объекта», которые формируют основу для объектов изображения и остальной части анализа изображения. Сегментация, а также результирующие характеристики примитивных и конечных объектов изображений основаны на форме, размере, цвете и топологии пикселей, контролируемые с помощью параметров, заданных пользователем. Значения параметров определяют насколько сильно будут влиять спектральные и пространственные

характеристики слоев изображения на определение формы и размера интересующих объектов.

Методы сегментации разбиваются на два вида: контекстуальные, неконтекстуальные. Последние не учитывают пространственные отношения между объектами на изображениях и группируют пиксели вместе на основе некоторого глобального атрибута, например, уровня серого или цвета. Контекстуальные методы используют эти отношения и группируют вместе пиксели с одинаковыми серыми уровнями и близкими пространственными положениями [14].

Неконтекстуальное пороговое значение. Определение порога – это самый простой метод неконтекстной сегментации. Порог – это преобразование полутонового или цветного изображения в двоичное, рассматриваемое как двоичная матрица регионов. Бинарная матрица состоит из двух непересекающихся областей, одна из которых содержит пиксели со значениями входных данных, меньшие порогового значения, и другие, относящиеся к входным значениям, то есть на уровне или выше порога. Первый и последний регион обычно маркируется нулевой (0) и ненулевой (1) меткой соответственно. Результат сегментации зависит от самого изображения, являющегося обрабатываемым, и от того, как выбирается пороговое значение.

Как правило, неконтекстуальное пороговое значение может включать два или более порогов, а также производить более двух типов областей таких, что диапазоны входных сигналов изображения, связанные с каждым типом региона, разделены пороговыми значениями. Вопрос о пороговом значении заключается в том, как автоматически определяется пороговое значение.

Наиболее распространенным способом для нахождения порога является уровень серого цвета пикселей (формула 1).

$$g(x, y) = 0, \text{ если } f(x, y) < T \text{ и } g(x, y) = 1, \text{ если } f(x, y) \geq T, \quad (1)$$

где  $T$  – пороговое значение.

С помощью двух порогов,  $T_2 < T_1$ , диапазон уровней серого цвета связанные с первой областью можно определить по формуле 2.

$$\begin{aligned} g(x,y) &= 0, \text{ если } f(x,y) < T_1 \text{ или } f(x,y) > T_2, \\ g(x,y) &= 1, \text{ если } T_1 \leq f(x, y) \leq T_2 \end{aligned} \quad (2)$$

Основные проблемы заключаются в том, возможно ли это и, если да, то как выбрать наиболее подходящий порог, чтобы отделить один или несколько желаемых объектов от фона. Во многих практических случаях простое пороговое значение, которое вычисляется по данным формулам, не может сегментировать интересующие объекты.

Общий подход к пороговым значениям основан на предположении, что изображения являются мультимодальными, то есть разные интересующие объекты представляют отдельные пики (или моды) гистограммы (рисунок 7). Порогом необходимо оптимально разделить эти пики, несмотря на типичные перекрытия между ними. Оптимальный порог, который минимизирует ожидаемое число ложных обнаружений и отклонений может не совпадать с ожидаемым числом ложных обнаружений и отклонений. Самая низкая точка между двумя вершинами является оптимальным порогом для рассматриваемого изображения.

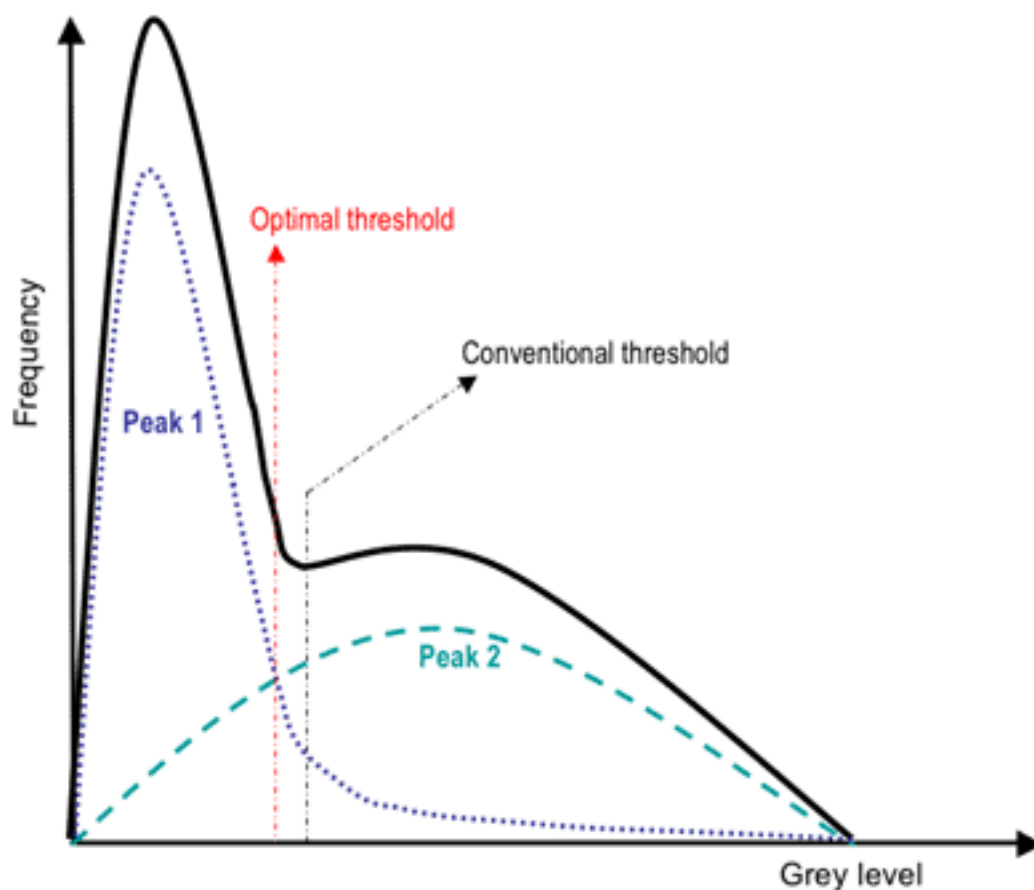


Рисунок 7 – Пример нахождения порога сегментации

Так как порог разделяет фон от объекта, адаптивное разделение может учитывать эмпирическую вероятность распределение объектов (темные) и фоны (яркие) пикселей. Такой порог должен уравновесить два вида ожидаемой погрешности присвоения пиксель фона для объекта и назначения пиксель объекта на заднем плане. Методы используют более сложные адаптивные пороговые значения, основанные на пространственно изменяющихся порогах, чтобы компенсировать локальные контекстные эффекты (такой пространственно изменяющийся порог можно рассматривать, как фоновую нормализацию).

Простая итеративная адаптация порога основана на последовательном уточнении предполагаемых пиковых положений. Он предполагает, что (i) – каждый пик совпадает со средним полутоновым уровнем для всех пикселей, что касается этого пика и (ii) – вероятность появления пикселя монотонно уменьшается на абсолютной разнице между пиксельными и пиковыми

значениями, как для объекта, так и для фонового пика. Классификация объекта и фоновых пикселей выполняется на каждой итерации  $j$  с помощью найденного порога  $T_j$  на предыдущей итерации. Таким образом, на итерации  $j$  каждый полутоновый уровень  $f(x, y)$  присваивается сначала объекту или фоновому классу (региону), если  $f(x,y) \leq T_j$  или  $f(x,y) > T_j$ , соответственно.

Таким образом, по формулам 3 и 4 находится адаптивный порог для полутонового изображения.

$$\mu_{j,ob} = \frac{\sum_{q=0}^{T_j} qh(q)}{\sum_{q=0}^{T_j} h(q)}; \quad \mu_{j,bg} = \frac{\sum_{q=T_j+1}^{255} qh(q)}{\sum_{q=T_j+1}^{255} h(q)}, \quad (3)$$

где  $\mu_{j,ob}$  и  $\mu_{j,bg}$  обозначают средний полутоновый уровень на итерации  $j$  для найденного объекта и фоновых пикселей.

$$T_{j+1} = 0,5(\mu_{j,ob} + \mu_{j,bg}) \quad (4)$$

где  $h$  – значения гистограммы изображения.

Сегментация текстуры происходит за счет вычисления спектральных характеристик. Текстура – это пространственное свойство, характеризующее группы пикселей [22]. Локальная мера измерения, текстура вычисляется по окрестности. Пример простейшей статистики меры – это дисперсия уровней полутонового цвета в квадратной окрестности  $n \times n$  центрированной на пикселе (формула 5).

$$\sigma^2 = \frac{1}{n^2} \sum_{\xi_{m-n/2}}^{n/2} \sum_{\eta_{m-n/2}}^{n/2} (f(x + \xi, y + \eta) - \mu)^2;$$

$$\sigma^2 = \frac{1}{n^2} \sum_{\xi_{m-n/2}}^{n/2} \sum_{\eta_{m-n/2}}^{n/2} (f(x + \xi, y + \eta) - \mu)^2 \quad (5)$$

Представленные масштабированные стандартные отклонения дисперсии  $\sigma$  для каждого пикселя: яркие области на изображении обозначают высокую локальную дисперсию уровней серого цвета:

Контекстная сегментация может добиться более высоких результатов в разделении объектов, потому что она учитывает расположение пикселей,



которые принадлежат к отдельному объекту. Два основных подхода к контекстуальной сегментации основаны на разрыве сигнала или подобии. Основанный методы на разрыве пытается найти полные границы, охватывающие относительно однородные области, предполагая резкие изменения сигнала через каждую границу. Методы, основанный на подобии, напрямую создают эти однородные области путем группировки пикселей, связанных объектов, удовлетворяющие определенным критериям подобия.

Метод связанности пикселей формирует арифметическую решетку:  $\{(x, y): x = 0, 1, \dots, X - 1; y = 0, 1, \dots, Y - 1\}$ . Есть два варианта окрестностей, которые возможно сформировать: четырех и восьми связанный путь. Четырех связанный путь охватывает верхний, нижний, правый и левый пиксель от центрального:  $\{(x - 1, y), (x, y + 1), (x + 1, y), (x, y - 1)\}$ . Восьмисвязанный путь аналогичен, но добавляет четыре диагональных пикселя:  $\{(x - 1, y - 1), (x - 1, y), (x - 1, y + 1), (x, y + 1), (x + 1, y + 1), (x + 1, y), (x + 1, y - 1), (x, y - 1)\}$ .

Метод активный контур применяется для выделения интересующих областей. Контурные – это совокупность точек, которая подвергается процессу интерполяции. Контурные модели описывают границы объекта или любые другие особенности изображения, чтобы сформировать параметрическую кривую или контур. Процесс интерполяции может быть линейным, сплайновым и полиномиальным, который представляет собой кривую на изображении. Активные контурные модели включают модель змеи, модель змеино-векторного потока градиента, модель воздушного шара и геометрические или геодезические контуры [10].

Модель змеи в основном работает для идентификации и очертания целевого объекта, рассматриваемого для сегментации. Он использует определенное количество предварительных знаний о контуре целевого объекта, особенно для сложных объектов. Активная модель змеи обычно настраивается с помощью сплайна, сфокусированного для минимизации энергии, сопровождаемой различными силами, управляющими изображением. Сплайн минимизации энергии направляет ограничительные силы и вытягивается с

помощью внутренних и внешних сил изображения на основе соответствующих контурных объектов. Змея эффективно работает со сложными целевыми объектами, разбивая фигуру на различные более мелкие части.

Энергия контура зависит от размера, положения и внешней, внутренней энергии рассматриваемого объекта. Контур выражен:  $v(s)=(x(s), y(s))$ . Энергия находится по формуле 6.

$$E[v(s)] = E_v[v(s)] + E_s[v(s)], \quad (6)$$

где  $E_v$  – внутренняя энергия,  $E_s$  – внешняя энергия.

Таким образом, для изображений, где уровень значений пикселей формирует контрастные области, контур вычисляется по формуле 7.

$$K(P) = K(x,y) = -\Delta^2 (GI) \quad (7)$$

где  $I(P)$  – интенсивность изображения в точке  $P$ ,  $I * G$  – перемножение пикселей с фильтром Гаусса.

Существует множество методов для повышения контраста изображений. Их разделяют на прямые методы, которые вводят меру контраста и стремятся улучшить ее, и непрямые, которые используют не задействованные участки динамического диапазона.

Один из прямых методов – адаптивный алгоритм. Это локальный способ повышения контраста. Метод подстраивается под каждую окрестность: преобразует регион в зависимости от значения пикселей (формула 8).

$$H_s(W(i,j)) = \frac{L_{max}(W(i,j)) - L_{min}(W(i,j))}{H_{max}(W(i,j))}, \quad (8)$$

где  $L_{max}$  – максимальное значение пикселей,  $L_{min}$  – минимальное значение пикселей.

Проанализировав методы сегментации и алгоритмы обработки изображения можно сделать вывод, что наиболее подходящим методом для решения задачи является алгоритм активных контуров. Его точность

сегментации превышает другие рассмотренные методы. Зачастую не прямые алгоритмы повышения контраста тратят на вычисление время, неприемлемое для потоковой обработки больших изображений. Поэтому целесообразнее применить прямой метод повышения контраста по дисперсии значений выборки.

## 2.2 Архитектура и методы обучения сверточной нейронной сети

Сверточные сети - это тип модели глубокого обучения для обработки данных, архитектура которой вдохновлена организацией визуальной коры млекопитающих, и предназначена для автоматического и адаптивного изучения пространственных иерархий объектов, от низкоуровневых до высокоуровневых шаблонов [33]. Математическая конструкция сети обычно состоит из трех типов слоев: свертки, подвыборки и полностью связанных слоев. Первые два, слои свертки и подвыборки, выполняют извлечение объектов, тогда как третий, полностью связанный слой, отображает извлеченные объекты в конечный результат, классифицируя его (рисунок 8).

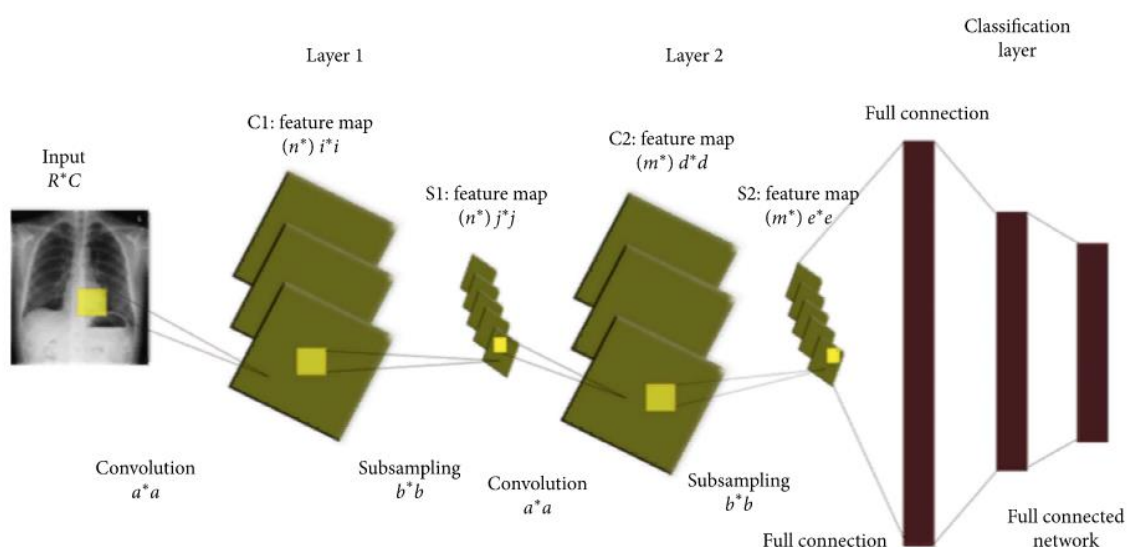


Рисунок 8 – Архитектура сверточной нейронной сети

Входной слой принимает на вход трехмерный массив, который задается параметром поступившего изображения (формула 9).

$$G = m*n*k, \quad (9)$$

где  $G$  – размер входного массива данных,  $m*n$  – размер изображения в пикселях (по вертикали и горизонтали соответственно),  $k$  – размерность массива, кодирующий цвет, в данном случае входные изображения представлены в полутоне, значит,  $k$  будет равно 1.

Далее изображение считывается на слой свертки, который является фундаментальным компонентом архитектуры CNN. Он выполняет извлечение объектов, которое обычно состоит из комбинации линейных и нелинейных операций, то есть операции свертки и функции активации.

Считывание происходит фильтром. Он представляют собой матрицу, которая проходит по изображению и производит математическую свертку. Результатом свертки является карта признаков, каждой пиксель которой означает степень схожести фрагмента изображения на фильтр.

Выходы линейной операции затем проходят через нелинейную функцию активации. Хотя гладкие нелинейные функции, например, сигмовидные или гиперболический тангенс функции были использованы ранее, потому что они являются математическими представлениями биологического поведения нейрона. Наиболее распространены нелинейные функции активации. Они используются для устранения линейных блоков. Вычисляется по простой функции:  $\Phi(X) = \max(0, x)$ .

Слой подвыборки обеспечивает типичную операцию понижающей дискретизации, которая уменьшает плоскостную размерность отображений объектов, чтобы ввести инвариантность преобразования к небольшим сдвигам и искажениям и уменьшить количество последующих изучаемых параметров. Следует отметить, что ни в одном из слоев нет изучаемого параметра, тогда как размер фильтра, шаг и заполнение являются гиперпараметрами в операциях объединения, аналогичными операциям свертки.

Выходные карты объектов конечного слоя свертки или подвыборки обычно сглаживаются, то есть преобразуются в одномерный массив чисел (или вектор) и соединяются с одним или несколькими полностью связными слоями, в которых каждый вход соединен с каждым выходом обучаемым весом. После создания объектов, извлеченных слоями свертки и понижающих выборок слоями объединения, они сопоставляются подмножеством полностью связанных слоев с конечными выходами сети, такими как вероятности для каждого класса в задачах классификации. Конечный полностью подключенный слой обычно имеет такое же количество выходных узлов, как и количество классов.

Когда происходит передача выходных данных из одного слоя в следующий слой, извлеченные объекты могут иерархически и постепенно усложняться. Процесс оптимизации параметров называется обучением, которое выполняется таким образом, чтобы минимизировать разницу между выходными и входными метками.

Обучение можно провести с помощью алгоритма обратного распространения ошибки или градиентного спуска.

Основная идея алгоритма обратного распространения ошибки состоит в распространении сигналов ошибки от выходов сети к её входам, в направлении, обратном прямому распространению сигналов в обычном режиме работы [7].

Метод градиентного спуска заключается в том, что функция потерь дает направление, в котором функция имеет самую большую скорость увеличения, и каждый обучаемый параметр обновляется в отрицательном направлении градиента с произвольным размером шага, определенным на основе гиперпараметра, называемого скоростью обучения [9].

### **2.3 Анализ моделей, предварительно обученных глубоких сверточных нейронных сетей**

Обучение глубокой нейронной сети занимает достаточно много времени, оно может длиться часы или дни, в зависимости от сложности архитектуры и размера набора данных. Также для этого требуется высокая вычислительная мощность. Один из вариантов решения данных проблем является использование предварительной обученной нейронной сети. Такие сети уже обучены классификации на миллионах изображениях и их возможно настроить под свою задачу.

В нейронной сети нейроны организованы слоями. Они могут выполнять различные виды преобразований на своих входных данных. Сигналы перемещаются от первого слоя (входного) до последнего (выходного) после многократного обхода слоев. Последний скрытый слой имеет достаточно обобщенной информации, чтобы обеспечить следующий слой, который выполняет фактическую задачу классификации. Конечная переподготовка слоев может работать на новых классах, так как вид информации, необходимый для различения всех классов, также возможен для различения новых видов объектов.

Модель VGG-16 была представлена на конкурсе ILSVRC Оксфордским университетом. Она обучена на наборе данных ImageNet, который состоит из более чем 15 миллионов размеченных изображений. Таким образом, данная сеть достигает 92,7% точности.

Архитектура модели VGG-16 представлена на рисунке 9. На вход подается изображение размера  $224 \times 224 \times 3$ . Первые два слоя модели имеют 64 канала, размеры фильтра которого  $3 \times 3$ . Пространственный пулинг осуществляется при помощи пяти max-pooling слоев, которые следуют за одним из сверточных слоев (не все сверточные слои имеют последующие max-pooling). Операция max-pooling выполняется на окне размера  $2 \times 2$  пикселей с шагом 2 [24].

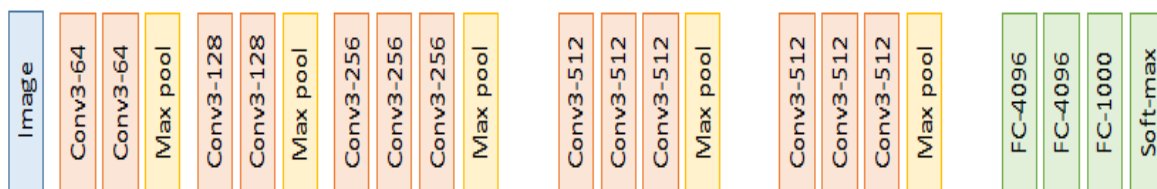


Рисунок 9 – Архитектура VGG-16

После стека свертки и слоя max-pooling формируется карта объектов  $7 \times 7 \times 512$ . Выход данных «сглаживается», чтобы получить вектор пространственных значений  $1 \times 25088$ . Далее идут три полносвязных слоя. Первый из них принимает выход от последней группы слоев и выводит вектор  $1 \times 4096$ . Второй также выводит вектор аналогичного размера, а последний формирует каналы той же размерности, что и количество классов. Этот вывод передается на последний слой softmax для нормализации вектора классификации. Все скрытые слои используют ReLU в качестве своей функции активации. ReLU является более эффективным с вычислительной точки зрения, поскольку она убыстряет процесс обучения, а также уменьшает вероятность схождения.

Модель нейронной сети VGG-19 является аналогичной в работе, как VGG-16 с разницей в том, что имеет на три слоя свертки больше (рисунок 10).

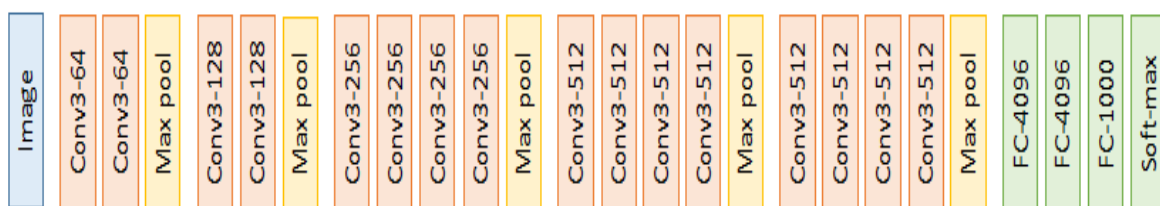


Рисунок 10 – Архитектура VGG-19

Данная модель обладает точностью выше – 95,7%, благодаря большему количеству слоев. Но данные две модели имеют схожую частоту ошибок. Если добавить в данную модель еще слоев, то точность снизится.

Модели VGG легки в реализации и надстройки весов и не требуют высокой вычислительной мощности.

Модель Inception-v3 обеспечивает наиболее высокую точность по сравнению с рассмотренными моделями. Она состоит из двух частей:

- извлечение признаков с помощью сверточных слоев;
- классификация полносвязными и softmax слоями.

На рисунке 11 изображена архитектура модели.

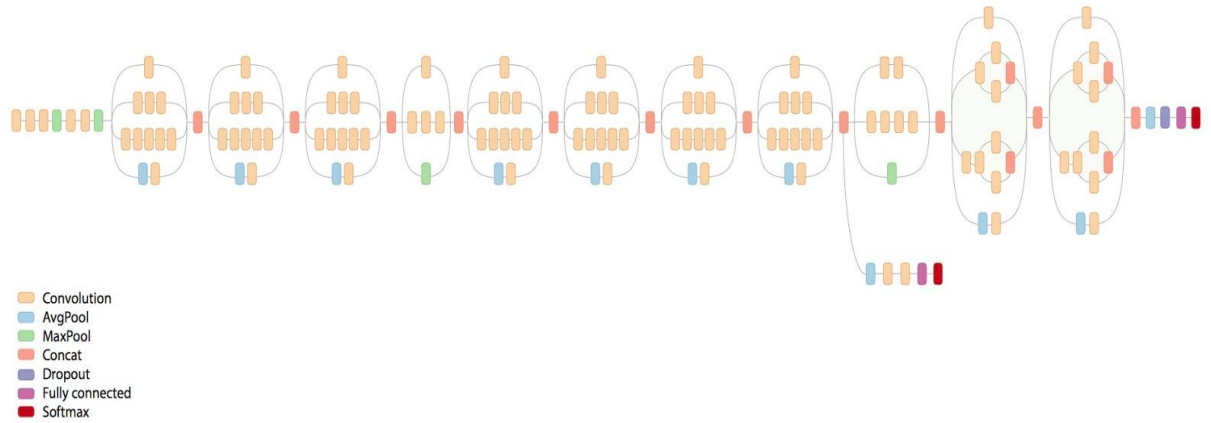
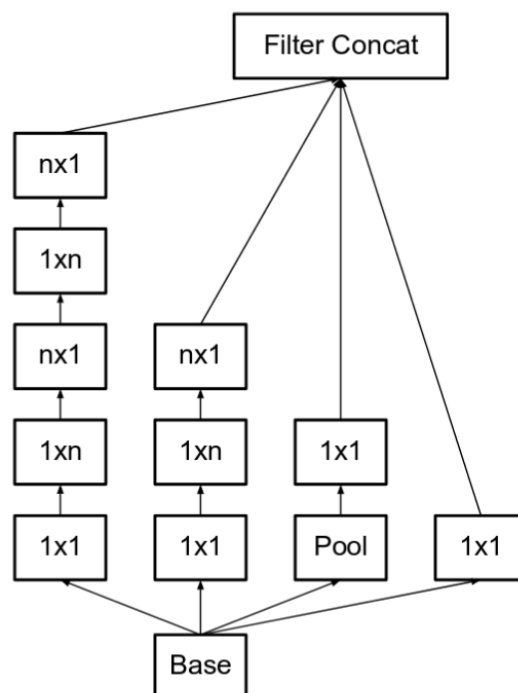


Рисунок 11 – Архитектура Inception-v3

Архитектура основывается на совокупности комбинации фильтров (рисунок 12).





## Рисунок 12 – Комбинация фильтров Inception-v3

Особенностью модели является параллельная комбинация сверточных каскадов размерностью  $1 \times 1$ ,  $3 \times 3$ ,  $5 \times 5$  и bottleneck-слоем. Данный слой уменьшает размер свойств изображений, таким образом, достигая высокой скорости вычисления.

По сравнению с предыдущими версиями модель снижает вычислительную сложность операции свертки, достигая за счет замены фильтра размерностью  $n \times n$ , последовательностью одномерных сверток размером  $n \times 1$  и  $1 \times n$  [12].

В таблице 4 представлены официальные экспериментальные результаты рассмотренных моделей [33].

Таблица 4 – Экспериментальные результаты

Нейронная сеть	Ошибка	Точность
VGG-16	9,2%	92,7%
VGG-19	8,6%	94,5%
Inception-v3	3,2%	97,8%

Наиболее высокие результаты из рассмотренных моделей показала нейронная сеть Inception-v3. Модель поможет облегчить решение задачи исследования.

### 2.4 Структура математической модели сверточной нейронной сети

Для описания математической модели глубокой сверточной нейронной сети используются следующие обозначения.

Под  $l \in [1; L]$  понимается рассматриваемый в данный момент слой нейронной сети, где  $L=2a+2$ ,  $a \in \mathbb{Z}$  - количество слоев в сети. За  $N^l$  обозначается количество карт признаков на слое  $l$ , а за  $f(\cdot)$  - функцию активации

рассматриваемого слоя  $l$ . Также под переменной  $y^l$  понимается  $n$ -ая карта признаков на слое  $l$ .

Математическая модель сверточного слоя. В рассмотрение вводится сверточный слой  $l$ . В подобной архитектуре нейронной сети принимается нечетным числом, то есть  $l = 2a - 1$ . Тогда для карты признаков  $n$  будет иметь место следующее:

1.  $w_{m,n}^l = \{w_{m,n}^l(i,j)\}$  свертка, применяемая к карте признаков  $m$  слоя  $(l-1)$ , на слое  $l$  с картой признаков  $n$ ;
2.  $b_n^l$  - пороговые значения, присоединяемые к карте признаков  $n$  на слое  $l$ ;
3.  $V_n^l$  - список всех уровней слоя  $(l-1)$ , которые соединяются с картой признаков  $n$  слоя  $l$ .

Таким образом, карта признаков  $n$  сверточного слоя  $l$  будет вычисляться формулой 10 [6].

$$y_n^l = f\left(\bigcup_{m \in V_n^l} y_m^{l-1} \otimes w_{m,n}^l + b_n^l\right) \quad (10)$$

Предполагается, что размер входных карт признаков  $y^{l-1}$  равен  $H^{l-1} \times W^{l-1}$ , а размер применяемой к ним свертки  $w_{m,n}^l$  равняется  $r^l \times c^l$ , тогда размер выходной карты признаков  $y_n^l$  вычисляется по формуле 11.

$$(H^{l-1} - r^l + 1) \times (W^{l-1} - c^l + 1) \quad (11)$$

Математическая модель слоя подвыборки. Вводится в рассмотрение слой подвыборки  $l$ . В сверточной нейронной сети принято принимать четным числом, то есть  $l=2a$ . Для карты признаков  $n$  вводится следующие обозначения:  $w_{m,n}^k$  - фильтр, применяемый к  $n$  на слое  $l$ , и  $b_n^l$  - добавочное пороговое значение.

Далее будут произведены следующие действия: разделение карты признаков  $n(l-1)$ -ого слоя на непересекающиеся блоки размером  $2 \times 2$  пикселя. Затем суммирование значения четырех пикселей в каждом блоке и в результате будет получена матрица  $Z^{l-1}_n = \{z^{l-1}_n(i,j)\}$ , элементами которой будут являться

соответствующие значения сумм. Таким образом, формула для вычисления значений элементов матрицы представлена в формуле 12.

$$z_n^{l-1} = y_n^{l-1}(2i - 1, 2j - 1) + y_n^{l-1}(2i - 1, 2j) + y_n^{l-1}(2i, 2j - 1) + y_n^{l-1}(2i, 2j) \quad (12)$$

Карта признаков подвыборочного слоя вычисляется по формуле 13.

$$y_n^l = f_l(z_n^{l-1} \times w_{m,n}^l + b_n^l) \quad (13)$$

Таким образом, возможно посчитать размер  $H^l \times W^l$  карты признаков  $y_n^l$  подвыборочного слоя (формула 14).

$$H^l = \frac{H^{l-1}}{2}, W^l = \frac{W^{l-1}}{2} \quad (14)$$

Математическая модель выходного слоя. Рассматривается выходной слой  $L$ , состоящий из единичных нейронов. Берется за  $N^L$  количество нейронов на данном слое. Как и при рассмотрении предыдущих слоев, обозначается за  $w_{m,n}^L$  - фильтр, применяемый к карте признаков  $m$  последнего сверточного слоя для получения перехода к нейрону  $n$  выходного слоя. Пусть  $b_n^L$  - пороговое значение, добавляемое к нейрону  $n$ .

Пользуясь введенными обозначениями, получена формула для подсчета значения выходного нейрона (формула 15).

$$y_n^l = f_l(\sum_{m=1}^{N^{l-1}} y_m^{l-1} w_{m,n}^l + b_n^l) \quad (15)$$

Таким образом, выходом сверточной нейронной сети является вектор (формула 16).

$$y = [y_1^l, y_2^l, \dots, y_{N^l}^l] \quad (16)$$

Таким образом, цель – минимизировать функцию потерь (формула 17) за счет выбора настраиваемых параметров обучаемой системы  $w$ .

$$E(w) = -d_n^p \cdot \log y_n^p \quad (17)$$

$E(w)$  представляет собой арифметическое среднее функций вида, которое представлено в формуле 18.

$$E^p = D(D^p, F(w, Z^p)), \quad (18)$$

где  $Z^p$  –  $p$ -ый вход системы,  $D^p$  – несоответствие между желаемым выходом системы для  $Z^p$  и значением, которое выдает система.  $E(w)$  будет вычисляться над обучающим множеством пар вида:  $\{(Z^1, D^1), \dots, (Z^p, D^p)\}$ .

Математическая модель данных, используемая для обучения и тестирования, будет представлена в виде множества, которое необходимо классифицировать. В свою очередь классификация решает следующую задачу. Задано конечное множество классов и имеется множество объектов, для конечного подмножества которых известно к какому классу они относятся. Это подмножество называется обучающей выборкой.

Классовая принадлежность остальных объектов не известна. Требуется построить алгоритм, способный классифицировать произвольный объект из исходного множества.

Классифицировать объект – значит указать номер (или наименование класса), к которому относится данный объект.

Формальная постановка задачи.

$S = \{s_1, \dots, s_n\}$  – множество рентгенографических снимков грудной клетки. Каждый снимок  $s \in S$  представляет собой последовательность пикселей  $M_s = (m_{11}, \dots, m_{1n}; m_{21}, \dots, m_{2n}; \dots; m_{n1}, \dots, m_{nn})$ ,  $n_s$  – размер снимка  $s$ .

$Y = \{y_1, \dots, y_n\}$  — конечное множество меток классов.

$y^*: D \rightarrow Y$  — неизвестная целевая зависимость, значения которой известны только на объектах конечной обучающей выборки  $S^m = \{(s_1, y_1), \dots, (s_m, y_m)\}$ .

Требуется построить алгоритм  $a: S \rightarrow Y$ , способный классифицировать произвольный объект  $s \in S$ .

Таким образом, сформирована математическая модель архитектуры сверточной нейронной сети и множества данных снимков, которые будут использоваться для обучения и тестирования нейронной сети на точность. Также сформулированы цели для верного построения нейронной сети.

## Выводы к главе 2

1) Метод активных контуров обладает всеми необходимыми параметрами для формирования улучшения гибридного метода сегментации рентгенографического снимка в виде уточнения краев сегментированных областей.

2) Прямой адаптивный метод повышения контраста поможет детализировать области на рентгенограммах, которые предположительно содержат очаги заболеваний, тем самым повысить концентрацию внимания на этих участках для нейронной сети.

3) Наиболее действенным способом классификации при небольшом объеме выборки являются модели нейронных сетей уже заведомо обученных. Также данный способ уменьшает время на обучение и вычислительные требования.

4) Архитектура нейронной сети Inception-v3 и VGG-16 может послужить основой в формировании глубокой сверточной нейронной сети для классификации рентгенографических снимков.

5) Произведено математическое моделирование сверточной нейронной сети для глубоко понимая построения архитектуры и впоследствии произведения надежной надстройки сети.

## **Глава 3 Формирование составной глубокой сверточной нейронной сети для задачи классификации рентгенографического снимка**

### **3.1 Обработка рентгенографических снимков**

Применение методов глубокого обучения к радиологическим изображениям стремительно расширяется, но одной из проблем в применении нейронных сетей к рентгенографическим снимкам является получение достаточного количества набора данных. Зачастую трудно для исследования получить доступ к изображениям за пределами своего учреждения из-за проблем конфиденциальности пациентов.

Предварительная обработка изображений и формирование базы данных для обучения, проверки и тестирования играют важную роль в создании корректно работающей сверточной нейронной сети. Размер и структура набора данных влияют на обучение нейронной сети. Для успешного формирования сети необходимо иметь базу с разнообразными данными, что позволит снизить вероятность переобучения и повысить процент точности результата. Для облегчения и ускорения процесса нахождения признаков на изображении применяются алгоритмы сегментации, которые позволяют отсеять неинформативные области, и фильтры, концентрирующие внимание на определенных участках за счет увеличения контраста.

#### **3.1.1 Обработка формата DICOM**

Рентгенографические снимки грудной клетки, которые предоставлены городской больницей, находятся в формате DICOM. Данный формат содержит послойные кадры рентгенографического снимка, извлечение которых позволит увеличить набор данных. Поэтому необходимо произвести конвертирование в формат JPEG, который поддерживает полутоновое изображение и минимально влияет на снимок, сохраняя детали, необходимые для распознавания. Размер изображения составляет 3046×2291. Если набор данных будет состоять из снимков такой размерности, то процесс обучения может длиться

продолжительное время, а также потребуется высокая вычислительная мощность. Поэтому целесообразнее будет уменьшить размер изображения в 11 раз.

При помощи шкалы Хаунсфилда производится отражение кадра органа в зависимости от рентгеновской плотности тканей. В данной задаче денситометрические показатели, которые будут использоваться представлены в таблице 1. Таким образом, возможно выделить области, которые представляют важность для исследования.

Объем базы данных составляет 547 рентгенографических снимков, содержащих норму – 96 снимков, пневмонию – 85 снимков, туберкулез – 76 снимков и другие патологии – 290 снимков. Данная база является недостаточно обширной для обучения нейронной сети, и в итоге может быть достигнут низкий процент точности результата, поэтому следует расширить набор снимков при помощи геометрических преобразований.

С помощью языка Python и пакета Pydicom возможно конвертирование формата DICOM в формат JPEG, а также отделение кадров по шкале Хаунсфилда, что способствует увеличению объема набора данных. Преобразование снимков способствует сохранению достаточного количества деталей, чтобы уверенно классифицировать изображения при одновременном уменьшении размеров файлов входных данных для снижения вычислительных требований и использования памяти.

Через функцию `dicom.read_file()` происходит чтение файла. В переменную `DicFil` записывается выбранный снимок. Далее необходимо вычислить общий размер трехмерного массива, с помощью команды `PixDim = (int(DicFil.Rows), int(DicFil.Columns), len(istFilesDCM))`. Переменная `PixDim` содержит размер изображения и число изображений. Далее по кадру считываются в массив пиксели изображений с помощью команды `PixSpac = (float(DicFil.PixelSpacing[0]), float(DicFil.PixelSpacing[1]), float(DicFil.SliceThickness))`. Таким образом, в переменной `PixSpac` содержится интересное изображение. Преобразование изображений по шкале



Хаунсфилда происходит при помощи отсеивания значений пикселей по значениям из таблицы 1.

Далее необходимо произвести конвертирование изображения в формат JPEG. Для этого используется пакет ruIab.

Результатом алгоритма является три снимка грудной клетки, сформированные в различных представлениях по шкале Хаунсфилда, в формате PNG (рисунок 13)

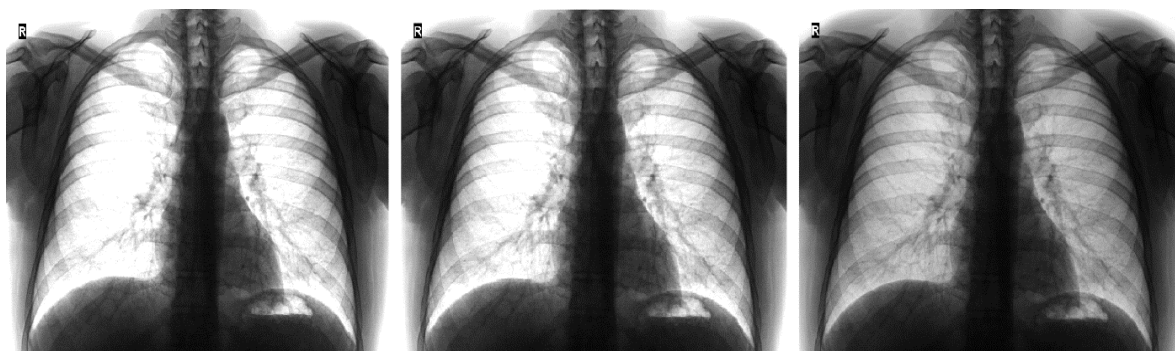


Рисунок 13 – Результат работы алгоритма

Таким образом, производится конвертирование рентгенографических снимков без потери информации и выделение кадров по шкале Хаунсфилда.

### 3.1.2 Сегментация рентгенографического снимка

В бакалаврской работе «Разработка и реализация метода сегментации рентгенограммы с наименьшим коэффициентом ошибки» рассматриваются четыре алгоритма сегментации рентгенографического снимка: «водораздел», бинаризация, однопроходный метод, «лесной пожар» [11]. В результате реализации методов выясняется их эффективность, которая представлена в таблице 5.

Таблица 5 – Эффективность методов сегментации

Наименование метода	Процент верно выделенной площади области легких
---------------------	---

Продолжение таблицы 5

«Водораздел»	20%
Бинаризация	80%
Однопроходный метод	60%
«Лесной пожар»	50%

По полученным сведениям, делается вывод, что целесообразнее составить гибридный алгоритм на основе бинаризации с варьируемым порогом и однопроходным методом. В итоге тестирования выявляется, что разработанный алгоритм верно выделяет 88% площади легочных долей.

Гибридный алгоритм подходит для данной задачи, так как выбранные заболевания не влияют на строение легочного рисунка, поэтому нет необходимости в соблюдении точности сегментации, и возможно произвести обработку без потери важных признаков. Результат работы гибридного алгоритма представлен на рисунке 14.

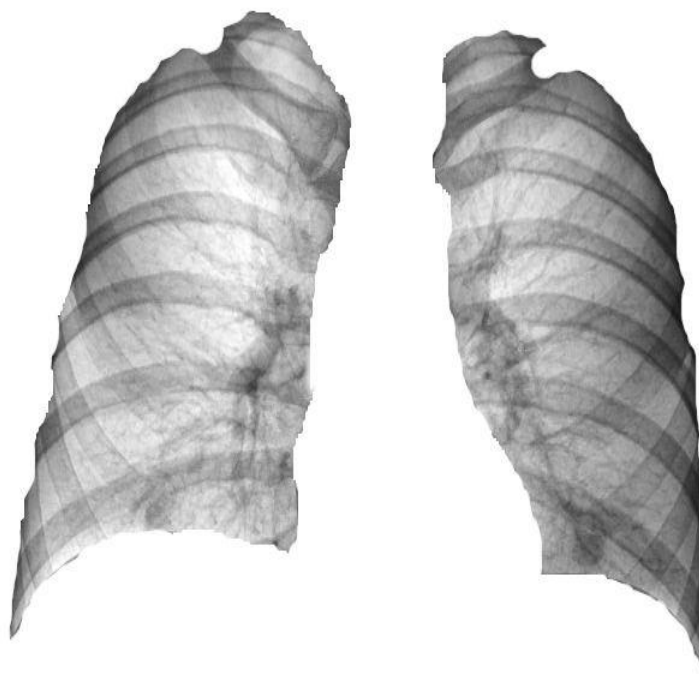


Рисунок 14 – Результат работы гибридного алгоритма

Для улучшения анализа снимка предлагается использовать метод активных контуров для уточнения и сглаживания границ легочных областей. Также локальное повышение контраста. Таким образом, в этих областях, которые были сегментированы на предыдущем шаге, будет повышен контраст для детализации, от чего патологии явно выделятся на снимке.

Очаги заболеваний туберкулеза и пневмонии преимущественно находятся в верхнем и среднем отделе легочной доли. Таким образом, повышается акцент на областях, предположительно содержащих патологию.

Данный алгоритм сегментации может работать только с рассматриваемыми заболеваниями, так как точный контур легочных долей не играет важности.

При помощи библиотеки OpenCV, которая позволяет производить преобразования над изображениями формулы 7 и 8 производится уточнение границ легочной структуры и повышение контраста.

Результат работы алгоритма представлен на рисунке 15.

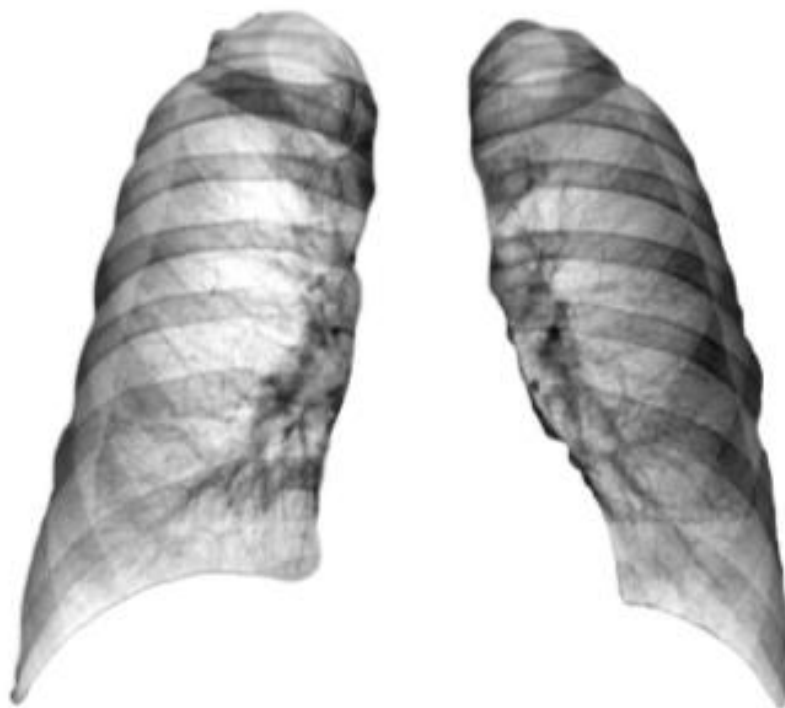


Рисунок 15 – Результат повышения контраста областей с патологией

Таким образом, производится уточнение границ легочной структуры и повышение контраста сегментированных областей легки, для улучшения четкости и детализации, что будет способствовать повышению точности.

### **3.2 Подготовка выборки для нейронной сети**

Обучающий набор данных используется для обучения нейронной сети, тестирования и проверки результата. Он включает в себя как входные данные, так и ожидаемые выходные. Обучающие наборы составляют большинство от общего количества данных, около 60%. При тестировании модели параметры подгоняются в процессе, известном как регулировка веса.

Процесс обучения направлен на выявление скрытых зависимостей и закономерностей в данных, которые будут проанализированы. Поэтому набор обучающих и тестовых данных должен представлять собой репрезентативную выборку целевых данных. Он должен быть достаточно широким, чтобы охватить все случаи.

Набор тестовых данных используется для оценки того, насколько хорошо алгоритм был обучен с помощью набора обучающих данных. Тестовые наборы представляют собой 16% данных. Набор состоит из маркированных данных из заведомо известных результатов.

База данных для задачи классификации заболеваний представляет собой набор рентгенографических снимков с метками патологий. Имеются из городской больницы 547 снимков и датасет chest X-Ray images, состоящий из 4518 снимков.

Необходимо произвести исследование: с помощью каких данных возможно повысить точность результата работы нейронной сети. Для обучения нейронной сети будет использоваться три разных набора данных:

1. снимки грудной клетки без преобразований;
2. сегментированные снимки легких;
3. сегментированные снимки легких с увеличенным контрастом.

Для задачи классификации выбраны снимки четырех классов: здоровые, пневмония, туберкулез и содержащие другую патологию. Так как обучение нейронной сети проходит с учителем, необходимо иметь данные с заранее известными результатами, а также часть данных пойдет на проверку и тестирование точности результата.

Благодаря обработке снимков в формате DICOM, которая была рассмотрена в пункте 3.1.1, набор данных был увеличен в 3 раза. Также расширить набор данных возможно с помощью геометрических преобразований. Обучающий и проверочный наборы были дополнены путем применения преобразований: сдвиг изображения в четыре разные стороны, поворот на 10 градусов.

В базе в общем количестве находится 6159 снимок, 8% из которых понадобится для проверки и 8% для тестирования, а также к ним добавятся видоизмененные снимки (таблица 6). Группа снимков сортируется в три папки: train, val, test. В каждой из которых находятся папки с подписанными заболеваниями (метки классов).

Таблица 6 – Распределение набора данных

Вид снимка	Обучение	Проверка	Тестирование
Норма	1108	530	525
Пневмония	1690	805	800
Туберкулез	1646	785	780
Патология	731	350	345

Таким образом, сформированы три разных набора данных для обучения нейронной сети: снимки грудной клетки без преобразований; сегментированные снимки легких; сегментированные снимки легких с увеличенным контрастом.

### 3.3 Реализация сверточной нейронной сети

Для классификации патологий используется глубокая сверточная нейронная сеть. Для того чтобы сократить время на обучение, применяется уже предварительно обученная нейронная сеть. За основу берется модель VGG-16 и Inception-v3.

### 3.3.1 Реализация сверточной нейронной сети на основе модели Inception-v3

Нейронная сеть состоит из двух частей: сверточной и полносвязной (рисунок 16). Сверточная часть выделяет характерные признаки в изображениях, на их основе полносвязная часть определяет, какой именно объект находится на снимке.

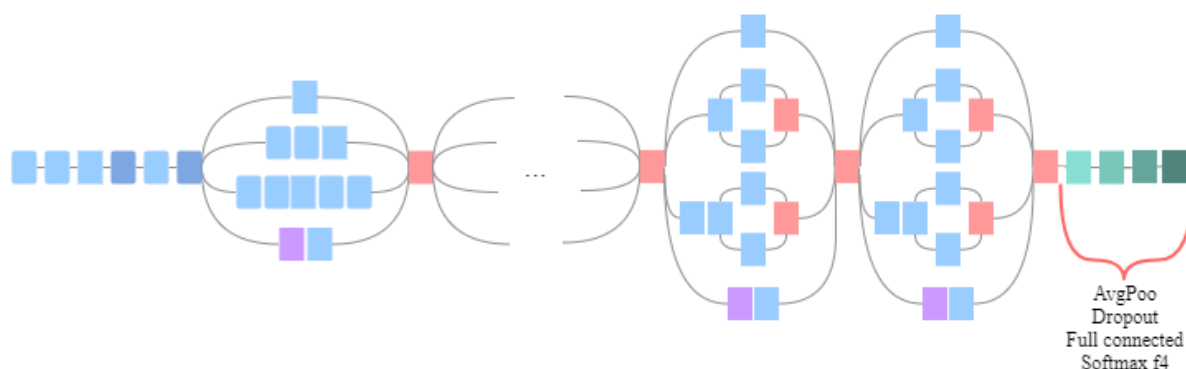


Рисунок 16 – Модель нейронной сети Inception-v3 с добавленной частью классификации

Для классификации патологий от модели Inception-v3 потребуется сверточная часть, к которой будет добавлена часть с определением типа объекта, состоящая из двух полносвязных слоев и имеющая на выходе четыре выходных нейрона (так как классифицируется только четыре вида снимка). Далее полученная нейронная сеть обучается на подготовленных наборах данных. Обученная модель и веса сохраняются для дальнейшего использования.

Перед моделированием нейронной сети необходимо создать генераторы изображений, которые сформируют выборки для обучения, проверки и тестирования.

Расширение наборов данных для тренировки и проверки происходит с помощью генератора функции `ImageDataGenerator()`. Каждое изображение поворачивается на 10 градусов – аргумент функции `rotation_range`, выполнение сдвига вправо, влево, вверх, вниз до 10% – `width_shift_range` и `height_shift_range` (рисунок 17).

```
#Преобразование изображений
train_datagen = ImageDataGenerator(rescale=1. / 255,
    rotation_range=10, #поворот
    width_shift_range=0.1, #сдвиг
    height_shift_range=0.1, #сдвиг
    fill_mode='nearest')
datagen = ImageDataGenerator(rescale=1. / 255)
```

Рисунок 17 – Листинг расширения набора данных

В переменную `train_generator` с помощью функции `flow_from_directory` помещается каталог снимков для обучения (`train_dir`), изображение преобразуется к размеру  $299 \times 299$  (`target_size`), размер мини-выборки на каждой эпохе составляет 32 (`batch_size`), и для каждого снимка формируется верный ответ по параметру `categorical` (рисунок 18).

```
#Генератор данных для обучения на основе
#изображений из каталога
train_generator = test_datagen.flow_from_directory(
    train_dir,
    target_size=(image_size, image_size),
    batch_size=batch_size,
    class_mode='categorical')
```

Рисунок 18 – Листинг генератор изображений для обучения

Аналогично создаются генераторы для проверки и тестирования с добавлением преобразованных изображений.

Для того чтобы узнать, к каким меткам присвоены классы снимков, используется команда `train_generator.class_indices`. Вызывая генератор изображений для обучения, он выводит индексы каждого класса. Таким образом: 0 – NORMAL, 1 – PNEUMONIA, 2 – TUBERCULOSIS, 3 – PATHOLOGY.

Предварительно загружается обученная нейронная сеть. В параметре `include_top` указывается то, что используется только сверточная часть. Далее отмечается, что не будет производиться обучение сверточных слоев и на данном этапе будут обучены только добавленные два полносвязных слоя. Это производится при помощи заморозки весов через `trainable` с атрибутом `False` (рисунок 19).

```
#Загрузка модели Inception-v3
pretrained_model = InceptionV3(weights='imagenet',
                                include_top=False)
#Заморозка весов
pretrained_model.trainable = False
```

Рисунок 19 – Листинг загрузки Inception-v3

Создается составная нейронная сеть на основе Inception. После добавляется слой, в котором с помощью `BatchNormalization()` происходит выпрямление данных – преобразование двумерных карт признаков из выхода сверточных слоев в вектор. Далее добавляется полносвязный слой с полулинейной функцией активации. Слой `Dropout`, что помогает отрегулировать снижение переобучения. Выходной полносвязный слой с четырьмя нейронами и функцией активации `softmax`, представляющей данные в виде числа вероятности принадлежности к классу (рисунок 20).



```

#Составная нейронная сеть. Добавляют полносвязные слои
x = pretrained_model.output
x = GlobalAveragePooling2D()(x)
x = Dense(512)(x)
x = BatchNormalization()(x)
x = Activation('relu')(x)
x = Dropout(0.5)(x)
predictions = Dense(3, activation='softmax')(x)
model = Model(inputs=pretrained_model.input,
              outputs=predictions)

```

Рисунок 20 – Листинг формирования составной нейронной сети

После задания нейронной сети необходимо ее скомпилировать, при этом задается низкая скорость обучения, так как сеть уже обучена, поэтому если задать скорость выше, может произойти переобучение. В качестве функции ошибки используется категориальная перекрестная энтропия (рисунок 21).

```

#Компилирование сформированной модели
model.compile(loss='categorical_crossentropy',
              optimizer='adam', lr=1e-5,
              metrics=['accuracy'])

```

Рисунок 21 – Листинг компиляции сформированной нейронной сети

Обучается сеть с помощью генераторов, которые предоставляют выборку для обучения и проверки. Обучение происходит в течение двадцати эпох (рисунок 22).

```

history = model.fit_generator(
    train_generator,
    steps_per_epoch=40,
    validation_data=test_generator,
    validation_steps=5,
    epochs=10,
    verbose=2)

```

Рисунок 22 – Листинг обучения нейронной сети

Процесс обучения на последних эпохах нейронной сети представлен на рисунке 23.

```

Epoch 00015:
  - 17s - loss: 0.0250 - acc: 0.9912 - val_loss: 0.7499 - val_acc: 0.7667
Epoch 16/20
4/4 [=====] - 1s 243ms/step - loss: 0.9100 - acc: 0.7500

Epoch 00016:
  - 17s - loss: 0.0222 - acc: 0.9941 - val_loss: 0.9100 - val_acc: 0.7500
Epoch 17/20
4/4 [=====] - 1s 238ms/step - loss: 0.5237 - acc: 0.8500

Epoch 00017:
  - 18s - loss: 0.0531 - acc: 0.9794 - val_loss: 0.5237 - val_acc: 0.8500
Epoch 18/20
4/4 [=====] - 1s 211ms/step - loss: 0.5649 - acc: 0.8583

Epoch 00018:
  - 18s - loss: 0.0160 - acc: 0.9956 - val_loss: 0.5649 - val_acc: 0.8583
Epoch 19/20
4/4 [=====] - 1s 212ms/step - loss: 0.5229 - acc: 0.8833

Epoch 00019:
  - 17s - loss: 0.0166 - acc: 0.9956 - val_loss: 0.5229 - val_acc: 0.8833
Epoch 20/20
4/4 [=====] - 1s 210ms/step - loss: 0.5804 - acc: 0.8833

Epoch 00020:
  - 16s - loss: 0.0317 - acc: 0.9897 - val_loss: 0.5804 - val_acc: 0.8833

```

Рисунок 23 – Эпохи обучения

Наиболее высокая точность результата показывается на последних эпохах. При увеличении периодов обучения может произойти переобучение сети.

Сохраняются полученные веса сети в файл формата .h5 и модель нейронной сети в формат .json, для того, чтобы использовать сеть в дальнейшем, не обучая ее каждый раз (рисунок 24).

```

#Сохранение, полученной модели
model_json = model.to_json()
json_file = open("vgg16.json", "w")
json_file.write(model_json)
json_file.close()

```

Рисунок 24 – Листинг сохранения модели нейронной сети и ее весов

Качество обучение выводится через evaluate\_generator, в которую предоставляется выборка из тестового генератора (рисунок 25).

```

scores = model.evaluate_generator(test_generator,
                                nb_test_samples // batch_size)
print("Точность на тестовых данных: %.2f%%" % (scores[1]*100))

```

Рисунок 25 – Листинг вывода точности на тестовых данных

Код моделирования и обучения составной нейронной сети представлен в приложении А.

Таким образом сформирована глубокая нейронная сеть на основе модели Inception-v3.

Для того чтобы улучшить работу нейронной сети, необходимо дообучить некоторые сверточные слои под задачу классификации рентгенографического снимка. Таким образом, сформированная нейронная сеть будет классифицировать признаки связанные с легочной структурой.

В новый файл загружается полученная обученная модель нейронной сети с ее весами (рисунок 26).

```

#Загрузка данных нейронной сети
json_file = open("inceptionv3.json", "r")
loaded_model_json = json_file.read()
json_file.close()
# Создание модели на основе данных
loaded_model = model_from_json(loaded_model_json)
#Загрузка весов
loaded_model.load_weights("weights_inceptionv3.h5")

```

Рисунок 26 – Листинг загрузки обученной нейронной сети

Далее размораживаются два последних сверточных слоя, и с ними компилируется нейронная сеть (рисунок 27).

```

#Компиляция нейронной сети с размороженными слоями
model.compile(loss='categorical_crossentropy',
              optimizer='adam', lr=1e-5,
              metrics=['accuracy'])

```

Рисунок 27 – Листинг компиляции нейронной сети с размороженными сверточными слоями

Обучение данных слоев происходит по тому же алгоритму и набору данных, что и обучение сети с новыми полносвязными слоями, представленными на рисунке 22, с разницей длины обучения, которая составляет пять эпох. Процесс обучения на последних эпохах представлен на рисунке 28.

```
Epoch 2/5
4/4 [=====] - 1s 207ms/step - loss: 0.5466 - acc: 0.8917

Epoch 00002:
- 10s - loss: 0.0154 - acc: 0.9971 - val_loss: 0.5466 - val_acc: 0.8917
Epoch 3/5
4/4 [=====] - 1s 234ms/step - loss: 0.5259 - acc: 0.8813

Epoch 00003:
- 14s - loss: 0.1290 - acc: 0.9809 - val_loss: 0.5259 - val_acc: 0.8813
Epoch 4/5
4/4 [=====] - 1s 204ms/step - loss: 0.5758 - acc: 0.8833

Epoch 00004:
- 19s - loss: 0.0163 - acc: 0.9941 - val_loss: 0.5758 - val_acc: 0.8833
Epoch 5/5
4/4 [=====] - 1s 213ms/step - loss: 0.6435 - acc: 0.8917

Epoch 00005:
- 11s - loss: 0.0174 - acc: 0.9926 - val_loss: 0.6435 - val_acc: 0.8917
```

Рисунок 28 – Эпохи дообучения сверточных слоев

Также полученные веса сохраняются в файл с другим именем в формате h5.

Таким образом, смоделирована и обучена глубокая сверточная нейронная сеть, основанная на модели Inception-v3, для классификации заболеваний грудной клетки на рентгенограммах, таких как пневмония и туберкулез.

### 3.3.2 Реализация сверточной нейронной сети на основе модели VGG-16

Нейронная сеть VGG-16 также состоит из двух частей: сверточной и полносвязной. Для классификации патологий потребуется сверточная часть, к которой будет добавлена часть с определением типа объекта, состоящая из двух полносвязных слоев и имеющая на выходе четыре выходных нейрона (так как

классифицируется только четыре вида снимка) (рисунок 29). Далее полученная нейронная сеть обучается на подготовленных наборах данных. Обученная модель и веса сохраняются для дальнейшего использования.

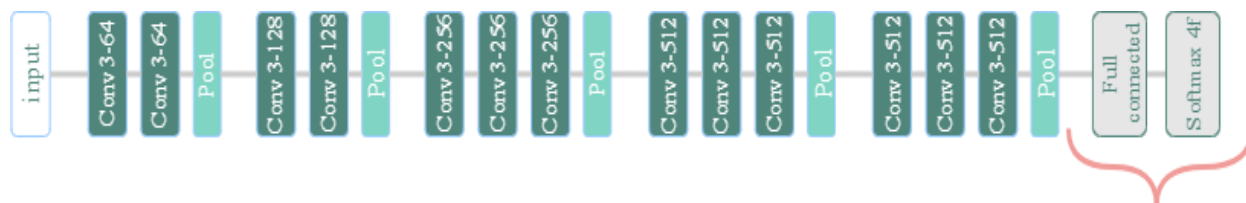


Рисунок 29 – Модель нейронной сети VGG-16 с добавленной частью классификации

Аналогично, как и для Inception-v3, предварительно формируются генераторы изображений для обучения, тестирования и проверки.

Предварительно загружается обученная нейронная сеть. В параметре `include_top` указывается то, что используется только сверточная часть. Далее отмечается, что не будет производиться обучение сверточных слоев и на данном этапе будут обучены только добавленные два полносвязных слоя (рисунок 30).

```
vgg16_net = VGG16(weights='imagenet',
                  include_top=False,
                  input_shape=(150, 150, 3))
vgg16_net.trainable = False
```

Рисунок 30 – Листинг загрузки VGG-16

Создается нейронная сеть в виде последовательной модели `Sequential()`, в которую добавляются загруженная часть сверточных слоев. После добавляется слой, в котором с помощью `Flatten()` происходит выпрямление данных – преобразование двумерных карт признаков из выхода сверточных слоев в вектор. Далее добавляется полносвязный слой с полулинейной функцией активации. Слой `Dropout`, что помогает отрегулировать снижение переобучения. Выходной полносвязный слой с тремя нейронами и функцией

активации softmax, представляющая данные в виде числа вероятности принадлежности к классу (рисунок 31).

```
model = Sequential()  
model.add(vgg16_net)  
model.add(Flatten())  
model.add(Dense(256))  
model.add(Activation('relu'))  
model.add(Dropout(0.5))  
model.add(Dense(3))  
model.add(Activation('softmax'))
```

Рисунок 31 – Листинг моделирования нейронной сети

После задания нейронной сети необходимо ее скомпилировать, при этом задается низкая скорость обучения, так как сеть уже обучена, поэтому если задать скорость выше, может произойти переобучение (рисунок 32).

```
model.compile(loss='categorical_crossentropy',  
              optimizer='adam', lr=1e-5,  
              metrics=['accuracy'])
```

Рисунок 32 – Листинг компиляция

Обучается сеть с помощью генераторов, которые предоставляют выборку для обучения и проверки. Обучение происходит в течении пяти эпох, так как при большом количестве эпох может произойти переобучение (рисунок 33).

```
model.fit_generator(  
    train_generator,  
    steps_per_epoch=nb_train_samples // batch_size,  
    epochs=5,  
    validation_data=val_generator,  
    validation_steps=nb_validation_samples // batch_size)
```

Рисунок 33 – Листинг обучения нейронной сети VGG-16

Аналогично сохраняется модель и веса полученной сети в файл формата .json и .h5 соответственно, для того, чтобы использовать сеть в дальнейшем, не обучая ее каждый раз (рисунок 24).

Код моделирования и обучения составной нейронной сети представлен в приложении Б.

Таким образом, была смоделирована и обучена сверточная нейронная сеть на основе модели VGG-16 для классификации заболеваний грудной клетки на рентгенограммах.

### Вывод к главе 3

1) Обработаны рентгенографические снимки грудной клетки формата DICOM. По денситометрическим показателям выделены три различных кадра снимка. Переведены в формат JPEG при минимальной потере информации и уменьшены в размере.

2) Рассмотрен гибридный алгоритм и сформировано дополнение к алгоритму сегментации. При помощи метода активных контуров уточнены и сглажены границы областей легких. С помощью адаптивного метода повышен контраст сегментированных областей с целью детализации и повышении качества.

3) Сформировано три набора данных: необработанные снимки; сегментированные снимки; сегментированные снимки с увеличенным контрастом. Объем данных для тестирования увеличен при помощи геометрических преобразований.

4) Реализованы глубокие сверточные нейронные сети на основе модели Inception-v3 и VGG-16. Добавлены два полносвязных слоя и дообучено несколько последних сверточных слоев с целью повышения точности нейронной сети.



## Глава 4 Проведение эксперимента на эффективность работоспособности алгоритма

### 4.1 Сравнительный анализ результатов классификаций

Для оценки качества классификации сформированных нейронных сетей используются метрики.

Метрика F-мера вычисляет один критерий качества на основе точности и полноты. Для вычисления критерия вводятся понятия *precision*, *recall*. *Precision* – доля объектов, названных классификатором положительными и при этом действительно являющимися положительными, а *recall* показывает, какую долю объектов положительного класса из всех объектов положительного класса нашел алгоритм (формула 19, 20).

$$precision = \frac{TP}{TP + FP}, \quad (19)$$

$$recall = \frac{TP}{TP + FN}, \quad (20)$$

где TP – доля истинно-положительных ответов, FP – ложноположительная доля, FN – ложно-отрицательная доля.

F-мера является среднее гармонической *precision* и *recall* (формула 21), поэтому вычисленная оценка будет точнее отражать результативность системы, чем оценка точности при неравномерной выборке данных.

$$F_{\beta} = (1 + \beta^2) \cdot \frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall} \quad (21)$$

Значение  $\beta$  определяет вес полноты при  $\beta=1$ , при  $0 < \beta < 1$  – точность.

Если значение F-меры близко к единице, значит сформированный алгоритм показывает лучший результат.

Чувствительность (*sensitivity*) и специфичность (*specificity*) являются важными составляющими оценки классификатора. Чувствительность находится

по формуле 19 и равна precision. Специфичность – это доля правильно идентифицированных отрицательных случаев (формула 22).

$$specificity = \frac{TN}{TN + FP}, \quad (22)$$

где TN – доля верно-отрицательных случаев.

По данным метрикам возможно построить ROC-кривую. Данной кривой наиболее часто визуально представляют результаты исследования. Кривая представляет собой зависимость количества верно классифицированных положительных случаев от количества неверно классифицированных отрицательных случаев.

ROC-кривая строится по следующему алгоритму: для каждого значения выборки с шагом  $d$  рассчитывается значение чувствительности и специфичности. Далее строится график зависимости: по оси Y откладывается sensitivity, по оси X – recall. В результате должен быть получен график, содержащий кривую, свидетельствующую об эффективности классификатора, и прямая  $y=x$ , относительно которой оценивается построенная кривая. Чем вычисленная кривая ближе к левому верхнему углу, тем выше предсказательная способность модели. Если же кривая ближе или совпадает с прямой  $y=x$ , то менее эффективней модель. Прямая обозначает полную неразличимость классов классификатором. При сравнении двух классификаторов оценивается расположение кривых относительно друг друга, кривая, которая расположена выше и левее, эффективнее другой модели.

Сверточные нейронные сети обучаются на трех сформированных наборах данных:

1. снимки грудной клетки без преобразований;
2. сегментированные снимки легких;
3. сегментированные снимки легких с увеличенным контрастом областей.

Была получена оценка F-меры по результату обучения, которая внесена в таблицы 7, 8.

Таблица 7 – Средне гармонический критерий по видам снимков

Вид снимка	precision	recall	F-мера
Inception-v3			
Пневмония	0,97	0,93	0,95
Туберкулез	0,86	0,75	0,80
Норма	0,90	0,80	0,85
Патология	0,78	0,89	0,84
VGG-16			
Пневмония	0,87	0,79	0,82
Туберкулез	0,78	0,71	0,74
Норма	0,65	0,82	0,73
Патология	0,75	0,64	0,69

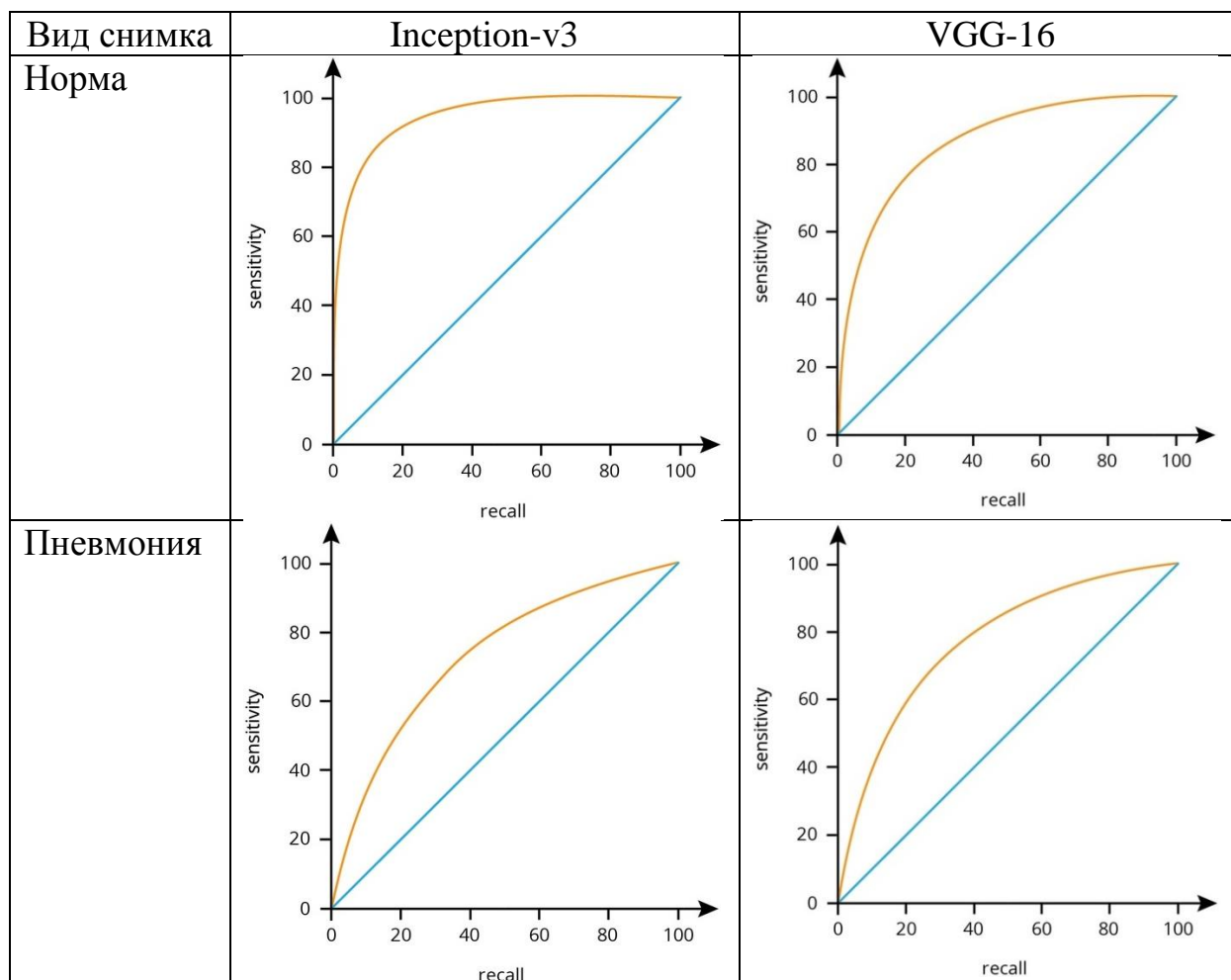
Таблица 8 – Средне гармонический критерий нейронных сетей, обученных на различных наборах данных

Набор данных	precision	recall	F-мера
Inception-v3			
Необработанные снимки	0,84	0,80	0,82
Сегментированные снимки	0,86	0,79	0,82
Сегментированные снимки с повышенным контрастом	0,89	0,84	0,86
VGG-16			
Необработанные снимки	0,70	0,63	0,66
Сегментированные снимки	0,74	0,68	0,71
Сегментированные снимки с повышенным контрастом	0,76	0,74	0,75

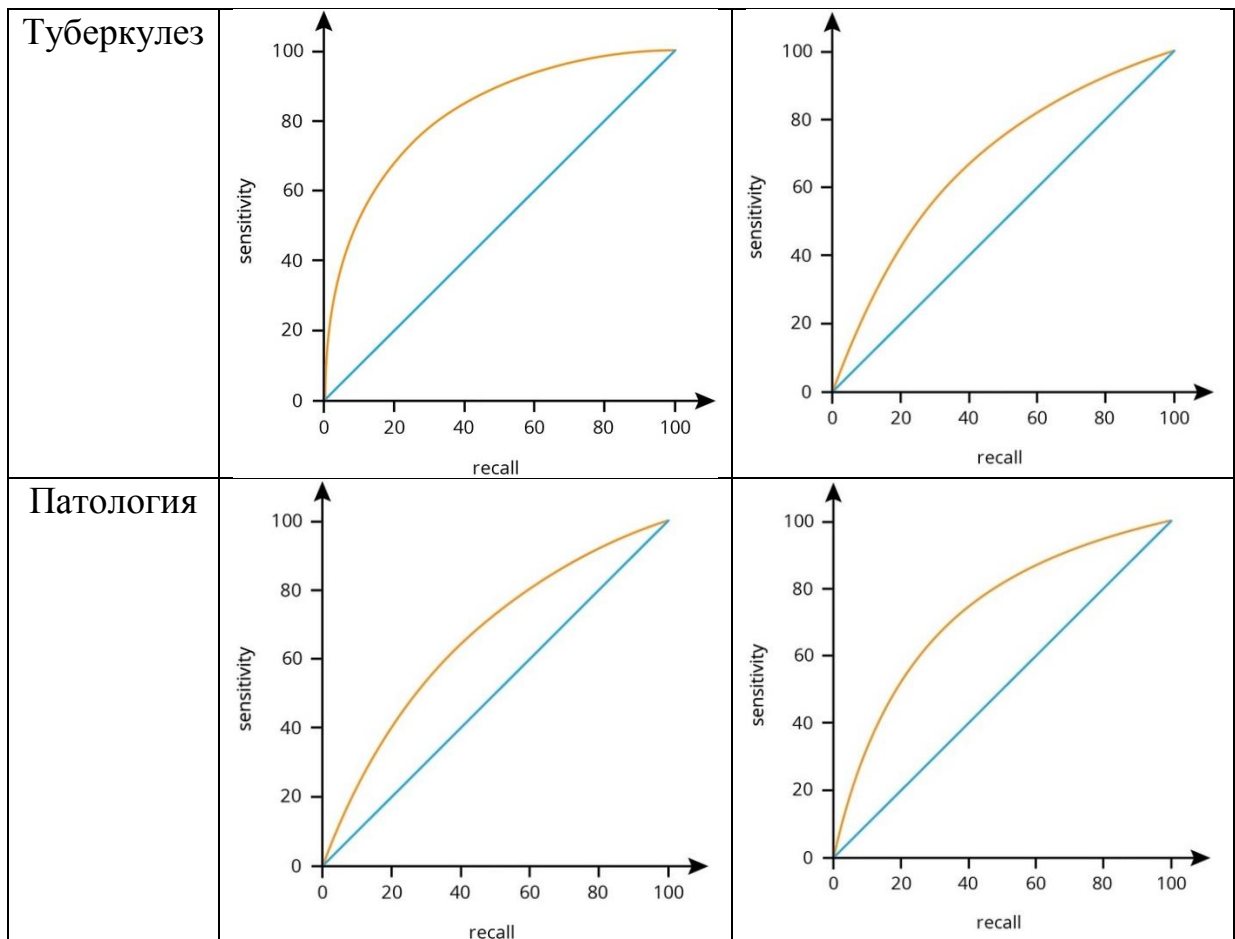
Из таблицы 8 видно, что наиболее влияющий на точность в положительную сторону, является набор данных, состоящий из сегментированных снимков с повышенным контрастом.

По вычисленным метрикам recall и sensitivity были построены ROC-кривые для анализа эффективности классификации нейронных сетей. В таблице 9 представлены ROC-кривые классификации различных видов снимков.

Таблица 9 – ROC-кривая по видам снимков



Продолжение таблицы 9



Кривая находится на достаточном расстоянии от прямой, что свидетельствует о том, что предсказательная способность сетей не сведена к нулю.

Для сравнительной оценки ROC-кривых используется метод вычисления площадей под кривыми – Area under curve (AUC), который находится по численному методу трапеции (формула 23).

$$AUC = \int f(x)dx = \sum_i \left[ \frac{X_{i+1} + X_i}{2} \right] \cdot (Y_{i+1} - Y_i) \quad (23)$$

В результате вычисления получены оценки AUC, которые сведены в таблицу 10.

Таблица 10 – Оценка AUC для ROC-кривых

Вид снимка	Inception-v3	VGG-16
------------	--------------	--------

Продолжение таблицы 10

Норма	91%	87%
Пневмония	78%	81%
Туберкулез	88%	70%
Патология	69%	76%

Результаты, сведенные в таблицы 7-10, указывают на то, что нейронная сеть, основанная на модели Inception, эффективнее и обладает хорошей точностью в классификации.

Таким образом, можно сделать вывод, что для достижения высокого результата точности следует использовать нейронную сеть на основе модели Inception-v3 и обучать на выборке, которая представляет собой сегментированные снимки с повышенным контрастом.

#### **4.2 Классификация рентгенографических снимков глубокой сверточной нейронной сетью**

Проверка работоспособности, полученной сверточной нейронной сети, производится на нескольких рентгенографических снимках, которые не использовались в обучении и тестировании.

Загружается модель и веса, полученной нейронной сети (рисунок 30). Формируется массив, содержащий название классов: «Норма», «Пневмония», «Туберкулез», «Патология», в которые нейронная сеть будет распределять изображения

Снимки загружаются в формате DICOM, приводятся к размеру, на котором была обучена сеть. Предварительная обработка изображения преобразует его в одномерный массив данных, который будет поступать на вход в нейронную сеть для классификации (рисунок 34).

```
#Преобразование изображения в массив
x = image.img_to_array(img)
x /= 255
x = np.expand_dims(x, axis=0)
```

### Рисунок 34 – Листинг предобработки входного снимка

Классификация снимка производится с помощью функции `predict()`, передавая в переменную `prediction` результат работы нейронной сети, который представляет собой одномерный массив с тремя значениями.

В качестве тестируемых рентгенографических снимках взяты четыре снимка, содержащих здоровые легкие, пневмонию, туберкулез и другую патологию.

В данном порядке снимки загружены в нейронную сеть, результат классификации представлен на рисунке 35.

```
[0.98536 0.01464 0.00000 0.00000]  
Класс: Норма
```

```
[0.08236 0.64170 0.09057 0.18537]  
Класс: Пневмония
```

```
[0.06581 0.20525 0.56512 0.35987]  
Класс: Туберкулез
```

```
[0.00739 0.32876 0.27143 0.39242]  
Класс: Патология
```

### Рисунок 35 – Результат классификации

Таким образом, проверена работоспособность сформированной глубокой сверочной нейронной сети. В результате не выявлено некорректного поведения программы и данные ею ответы соответствуют действительному.

## Вывод к главе 4

1) Произведено обучение нейронной сети на трех наборах данных: снимки грудной клетки без преобразований; сегментированные снимки легких; сегментированные снимки легких с увеличенным контрастом областей, содержащих патологии.

2) Выведены результаты точности нейронных сетей, произведена проверка по критерию F-меры и построены ROC-кривые.

3) По полученным точностям классификации и результатам проверки можно сделать вывод, что для достижения наиболее лучшего результата стоит использовать нейронную сеть на основе модели Inception-v3 и обучать на выборке, которая представляет собой сегментированные снимки с повышенным контрастом.

4) Произведена проверка на работоспособность сформированной нейронной сети. В результате классификации различных видов рентгенограмм нейронная сеть показала верные ответы.



## Заключение

В процессе работы над магистерской диссертацией были описаны актуальность и проблематика рассматриваемой темы, выявлен объект, предмет, поставлена цель и сформулированы задачи по теме исследования.

Проанализированы виды заболеваний органов легких и формы проявления на рентгенографическом снимке. Выбраны заболевания для последующей классификации: пневмония и туберкулез. Произведен обзор современного состояния существующих методов классификации рентгенограмм. Сделан вывод, что для выполнения цели исследования наиболее подходит заведомо обученная нейронная сеть. Проведен анализ, выбран и описан язык программирования, с помощью которого происходит реализация классификации, описан формат DICOM. Сформулированы требования и задачи для выполнения решения проблемы исследования.

Произведен анализ методов сегментации с целью улучшения гибридного алгоритма, реализованного в бакалаврской работе. В результате выбран метод активных контуров и адаптивный метод повышения контраста. Рассмотрена архитектура сверточной нейронной сети и произведен сравнительный анализ эффективности моделей заведомо обученных сетей. Сформирована математическая модель сверточной нейронной сети.

Реализованы алгоритм обработки формата DICOM и улучшен алгоритм сегментации рентгенографического снимка. Получены сегментированные области легких с повышенным контрастом регионов. Сформированы три набора данных для обучения нейронной сети. Реализованы глубокие сверточные нейронные сети на основе модели Inception-v3 и VGG-16, и произведена тонкая настройка для улучшения точности результата.

Полученные нейронные сети обучены на трех различных наборах данных. В результате сделан вывод, что целесообразнее использовать нейронную сеть на основе модели Inception-v3 и обучать ее на наборе, содержащим сегментированные снимки с увеличенным контрастом. Реализован

алгоритм работы с нейронной сетью и произведена проверка на работоспособность. В результате не выявлено некорректного поведения алгоритма. Алгоритм полностью соответствует поставленным требованиям.

Реализованная глубокая сверточная нейронная сеть достигает 89,17% точности классификации рентгенографических снимков, что является хорошим показателем сети, обученной на небольшом объеме данных.

Разработанный в процессе диссертационной работы алгоритм поможет в повышении эффективности диагностики заболеваний легких, как вспомогательный инструмент для врача, а также снизит коэффициент неверно поставленных диагнозов. Алгоритм имеет перспективу улучшения в повышении точности за счет увеличения набора данных и повышении качества постановки диагноза, учитывая другие факторы, влиявшие на пациента.

## Список используемых источников

### *Нормативно-правовые акты*

1. ГОСТ 19.701-90. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения [Текст]. Введ. 1992-01-01. – М. : Изд-во стандартов, 1992. – 14 с.
2. ГОСТ 2.105–95. Общие требования к текстовым документам [Текст]. – Введ. 1995-04-26. – М. : Госстандарт России: Изд-во стандартов, 1996. – 29 с.
3. ГОСТ 7.1-2003. Библиографическая запись. Библиографическое описание. Общие требования и правила составления [Текст]. – Введ. 2004-07-01. – М. : Госстандарт России: Изд-во стандартов, 2004.
4. ГОСТ 7.32-2001. Отчет о научно-исследовательской работе. Структура и правила оформления [Текст]. Введ. 2002-07-01. – М. : Госстандарт России: Изд-во стандартов, 2002.
5. ГОСТ 7.82-2001. Библиографическая запись. Библиографическое описание электронных ресурсов. [Текст]. Введ. 2001-05-01. – М. : Госстандарт России: Изд-во стандартов, 2001.

### *Научная и методическая литература*

6. Астафьева В.В. Разработка математической модели нейронной сети [Текст] / В.В. Астафьева ; Молодой ученый. - 2016. - №19. – 1-4 с.
7. Гузий Е.А. Сверточная нейронная сеть для разработки системы распознавания и классификации изображений [Текст] / Е.А. Гузий, В.В. Федоренко ; Молодежный научно-технический вестник. – 2017.
8. Лисовская, М.Г. Разработка алгоритма распознавания патологий на цифровых рентгенограммах [Текст] / М.Г. Лисовская; Научные исследования и перспективные проекты. – Тольятти, 2017. – 17-21 с.
9. Немков Р.М. Метод синтеза параметров математической модели сверточной нейронной сети с расширенным обучающим множеством [Текст] / Р.М. Немков ; Современные проблемы науки и образования. – 2015. - №1.
10. Семашко, А.С. Обработка и анализ границ объектов на основе метода

активных контуров [Текст] / А.С. Семашков, А.С. Крылов; МГУ имени Ломаносова, вычислительная математика и кибернетика, лаборатория математических методов обработки изображений. – Москва, 2015. – 4 с.

11. Спиркина А.А. Разработка и реализация метода сегментации рентгенограммы с наименьшим коэффициентом ошибки [Текст] / А.А. Спиркина; выпускная квалификационная работа. – [2018]. – 42 с.

12. Спиркина, А.А. Классификация патологий на цифровых рентгенограммах грудной клетки при помощи глубокой сверточной нейронной сети [Текст] / А.А. Спиркина, М.Г. Лисовская; сборник статей «Информационные технологии в моделировании и управлении: подходы, методы, решения». – Тольятти, 2020. – 276-282 с.

13. Талалаев А.А. Анализ эффективности применения искусственных нейронных сетей для решения задач распознавания, сжатия и прогнозирования [Текст] / А.А. Талалаев, И.П. Тищенко, В.П. Фраленко, В.М. Хачумов ; Искусственный интеллект и принятие решений. – [2008]. – № 2. – 24-33 с.

14. Тропченко А.Ю. Методы вторичной обработки и распознавания изображений : учебное пособие [Текст] / А.Ю. Тропченко, А.А. Тропченко. – СПб: Университет ИТМО. – [2015]. – 215 с.

15. Труфанов Г.Е. Лучевая диагностика : учебник [Текст] / Г.Е. Труфанов ; 8 глава. – [2014]. – 416 с.

16. Федорова А.А. Распознавание английского текста сверточной нейронной сетью [Текст] / А.А. Федорова ; Молодой ученый. – 2016. - №14. – 97-102 с.

17. Чучалин А.Г. Внебольничная пневмония у взрослых: практические рекомендации по диагностике, лечению и профилактике [Текст] / А.Г. Чучалин, А.С. Синопальников; Клиническая микробиология и антимикробная химиотерапия. – [2010]. – №3. – 12 с.

#### *Электронные ресурсы*

18. MATLAB [Электронный ресурс] / MATLAB. – Электрон. Дан. – [2018]. – Режим доступа: <https://matlab.ru/products/matlab>.

19. Python [Электронный ресурс] / Python. – Электрон. Дан. – [2016]. – Режим доступа: <https://all-python.ru/osnovy/yazyk-programmirovaniya.html>.

20. Баринаова, О. Методы сегментации изображений: автоматическая сегментация [Электронный ресурс] / О. Баринаова ; Графика. – Электрон. дан. – [2018]. – Режим доступа: <http://masters.donntu.org>.

21. Дементий, Д Язык программирования Java: особенности, популярность, ситуация на рынке труда [Электронный ресурс] / Д. Дементий ; Хекслет. – Электрон. дан. – [2019]. – <https://ru-hexlet-io.turbopages.org/s/ru.hexlet.io/blog/posts/yazyk-programmirovaniya-java-osobennosti-populyarnost-situatsiya-na-rynke-truda>.

22. Дюдин, М. В. Методы, модели и алгоритмы анализа и классификации растровых изображений рентгенограмм грудной клетки [Электронный ресурс] : учебное пособие / М. В. Дюдин ; Курск. – Вузовское образование, 2016. – 154 с.

23. Ланский, А. OpenCV – Python / А. Ланский; Tproger. – Электрон. дан. – [2019]. – Режим доступа: <https://tproger.ru/translations/opencv-python-guide/>

24. Милютин, И. VGG16 – сверточная сеть для выделения признаков изображений / И. Милютин; Neurogive. – Электрон. дан. – [2018]. – Режим доступа: <https://neurohive.io/ru/vidy-nejrosetej/vgg16-model/>

25. Обзор методов сегментации изображений в библиотеке scikit-image [Электронный ресурс] / Хабр. – Электрон. дан. – [2019]. – Режим доступа: <https://habr.com/ru/post/441006/>

26. Статистика по пневмонии и туберкулезу за 2018-20 гг. / Федеральная служба государственной статистики. – Электрон. дан. – [2020]. – Режим доступа: <https://www.gks.ru/>

#### *Литература на иностранном языке*

27. Abiyev, R.H. Deep convolutional neural networks for chest diseases detection / R.H. Abiyev; – Journal of Healthcare engineering. – 2018. – 11 pg. (нейронки)

28. Affiliations Author Lung nodules: improved detection with software that suppresses the rib and clavicle on chest radiographs / Author Affiliations; Radiology.

– [2011]. – №1. – 246 pg.

29. Eawat, W. Deep convolutional neural networks for image classification: a comprehensive review / W. Eawat, Z. Eang. – 2017.

30. Girshick, R. Region-based convolutional networks for accurate object detection and segmentation / Rose Girshick, Jeff Donahue, Trevor Darrell, J. Malik. – 2016.

31. Pelka, O. Annotation of enhanced radiographs for medical image retrieval with deep convolutional neural networks / O. Pelka, F. Nensa, C. Friedrich; – PLOS one. – 2018.

32. Soleymanpour Elaheh Full automatic lung segmentation and rib suppression methods to improve nodule detection in chest radiographs / Elaheh Soleymanpour ; Journal of medical signals and sensors. – [2011]. – 191-199 pg.

33. Szegedy, C. Rethinking the Inception Architecture for computer vision / C. Szegedy, V. Vanhoucke, S. Loffe, J. Shlens, Z. Wojna; Google Inc. – 2015. – 10 pg.

34. Teramoto Atsushi Automated classification of lung cancer types from cytological images using deep convolutional neural networks / Atsushi Teramoto ; BioMed Research International. – [2017]. – 6 pg.

35. Wan Ahmad Lung segmentation on standard and mobile chest radiographs using oriented Gaussian derivatives filter / Wan Siti Halimatul Munirah Wan Ahmad, W Mimi Diyana W Zaki ; BioMed Central. – 2015.

## Приложение А

### Код составной глубокой сверточной нейронной сети на основе модели

#### Inception-v3

```
1. train_dir = 'train'
2. test_dir = 'test'
3. val_dir = 'val'
4. input_shape = (image_size, image_size, 3)
5. train_datagen = ImageDataGenerator(rescale=1. / 255,
6. rotation_range=10,
7. width_shift_range=0.1,
8. height_shift_range=0.1,
9. fill_mode='nearest')
10. datagen = ImageDataGenerator(rescale=1. / 255)
11. test_datagen = ImageDataGenerator(rescale=1. / 255)
12. train_generator = test_datagen.flow_from_directory( train_dir,
13. target_size=(image_size, image_size), batch_size=batch_size,
14. class_mode='categorical')
15. test_generator = datagen.flow_from_directory( test_dir,
16. target_size=(image_size, image_size), batch_size=batch_size,
17. class_mode='categorical')
18. history = model.fit_generator( train_generator,
19. steps_per_epoch=40, validation_data=test_generator,
20. validation_steps=5, epochs=10, verbose=2)
21. scores = model.evaluate_generator(test_generator, nm_var // batch_size)
22. print("Точность на тестовых данных: %.2f%%" % (scores[1]*100))
23. model.save_weights("weights_inceptionv3.h5")
24. model.load_weights("weights_inceptionv3.h5")
25. pretrained_model.trainable = False
26. for layer in model.layers[:249]:
27. layer.trainable = False
28. for layer in model.layers[249:]:
29. layer.trainable = True
30. model.compile(loss='categorical_crossentropy',
31. optimizer='adam', lr=1e-5,
32. metrics=['accuracy'])
33. history = model.fit_generator(
34. train_generator,
35. steps_per_epoch=40,
36. validation_data=test_generator,
37. validation_steps=5,
38. epochs=5,
39. verbose=2)
40. model_json = model.to_json()
```

## Продолжение Приложения А

```
41. json_file = open("vgg16.json", "w")
42. json_file.write(model_json)
43. json_file.close()
44. scores = model.evaluate_generator(test_generator, nm_var // batch_size)
45. print("Точность на тестовых данных: %.2f%%" % (scores[1]*100))
model.save_weights("weights_inceptionv3.h5")
```



## Приложение Б

### Код составной глубокой сверточной нейронной сети на основе модели

#### VGG-16

```
1. train_dir = 'train'
2. val_dir = 'val'
3. test_dir = 'test'
4. img_width, img_height = 150, 150
5. batch_size = 32
6. input_shape = (img_width, img_height, 3)
7. datagen = ImageDataGenerator(rescale=1. / 255)
8. train_generator = datagen.flow_from_directory(
9. train_dir,
10. target_size=(img_width, img_height),
11. batch_size=batch_size,
12. class_mode='categorical')
13. val_generator = datagen.flow_from_directory(
14. val_dir,
15. target_size=(img_width, img_height),
16. batch_size=batch_size,
17. class_mode='categorical')
18. test_generator = datagen.flow_from_directory(
19. test_dir,
20. target_size=(img_width, img_height),
21. batch_size=batch_size,
22. class_mode='categorical')
23. print(train_generator.class_indices)
24. vgg16_net = VGG16(weights='imagenet', include_top=False,
input_shape=(150, 150, 3))
25. vgg16_net.trainable = False
26. model = Sequential()
27. model.add(vgg16_net)
28. model.add(Flatten())
29. model.add(Dense(256))
30. model.add(Activation('relu'))
31. model.add(Dropout(0.5))
32. model.add(Dense(3))
33. model.add(Activation('softmax'))
34. model.compile(loss='categorical_crossentropy',
35. optimizer='adam', lr=1e-5,
36. metrics=['accuracy'])
37. model.fit_generator(
38. train_generator,
39. steps_per_epoch=nb_train_samples // batch_size,
```

## Продолжение Приложения Б

```
40. epochs=5,  
41. validation_data=val_generator,  
42. validation_steps=nb_validation_samples // batch_size)  
43. scores = model.evaluate_generator(test_generator, nb_test_samples //  
batch_size)  
44. print("Точность на тестовых данных: %.2f%%" % (scores[1]*100))  
45. model_json = model.to_json()  
46. json_file = open("vgg16.json", "w")  
47. json_file.write(model_json)  
48. json_file.close()  
49. model.save_weights("weights.h5")
```