

Министерство науки и высшего образования Российской Федерации  
 федеральное государственное автономное  
 образовательное учреждение высшего образования  
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа информационных технологий и робототехники  
 Направление подготовки 09.04.04 Программная инженерия  
 Отделение школы (НОЦ) Отделение информационных технологий

### МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы <b>Способ сегментации лёгких на снимках КТ с использованием методов глубокого обучения</b>
---

УДК 004.932.1:621.179.152.1:615.849

Студент

Группа	ФИО	Подпись	Дата
8ПМ8И	Войцеховский Алексей Алексеевич		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Аксёнов С.В.	к.т.н.		

### КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Меньшикова Е.В.	к.ф.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Горбенко М.В.	к.т.н.		

### ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Губин Е.И.	к.ф.-м.н.		

Томск – 2020 г.

Министерство науки и высшего образования Российской Федерации  
 федеральное государственное автономное  
 образовательное учреждение высшего образования  
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа информационных технологий и робототехники  
 Направление подготовки (специальность) 09.04.04 Программная инженерия  
 Отделение школы (НОЦ) Отделение информационных технологий

УТВЕРЖДАЮ:  
 Руководитель ООП  
 \_\_\_\_\_ Губин Е.И.  
 (Подпись)    (Дата)    (Ф.И.О.)

**ЗАДАНИЕ  
на выполнение выпускной квалификационной работы**

В форме:

магистерской диссертации
--------------------------

(бакалаврской работы, дипломного проекта/работы, магистерской диссертации)

Студенту:

Группа	ФИО
8ПМ8И	Войцеховскому Алексею Алексеевичу

Тема работы:

Способ сегментации лёгких на снимках КТ с использованием методов глубокого обучения
---

Утверждена приказом директора (дата, номер)	№59-62/с от 28.02.2020 г.
---	---------------------------

Срок сдачи студентом выполненной работы:	10.06.2020
--	------------

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ:**

<p><b>Исходные данные к работе</b></p> <p><i>(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.).</i></p>	<p>Объектом исследования является способ сегментации лёгких на снимках компьютерной томографии на основе методов глубокого обучения.</p> <p>Для обучения модели доступны 267 изображений лёгких и размеченных масок с публичной платформы Kaggle.</p>
---	---

<p><b>Перечень подлежащих исследованию, проектированию и разработке вопросов</b>  <i>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе).</i></p>	<p>Аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; выбор алгоритмов глубокого обучения; выбор способов увеличения выборки; написание программного кода для обработки изображения и обучения моделей; обучение и тестирования моделей глубокого обучения на тестовых данных; обсуждение результатов выполненной работы; заключение работы.</p> <p>Дополнительно должны быть разработаны следующие разделы: финансовый менеджмент, ресурсоэффективность и ресурсосбережение; социальная ответственность; раздел на иностранном языке.</p>
--	--

<p><b>Перечень графического материала</b>  <i>(с точным указанием обязательных чертежей)</i></p>	
--	--

**Консультанты по разделам выпускной квалификационной работы**  
*(с указанием разделов)*

Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Меньшикова Екатерина Валентиновна
Социальная ответственность	Горбенко Михаил Владимирович
Раздел на иностранном языке	Пичугова Инна Леонидовна

**Названия разделов, которые должны быть написаны на русском и иностранном языках:**

Литературный обзор

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	28.02.2020
--	------------

**Задание выдал руководитель / консультант (при наличии):**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Аксёнов С.В.	к.т.н.		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8ПМ8И	Войцеховский Алексей Алексеевич		

**Министерство образования и науки Российской Федерации**  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа информационных технологий и робототехники  
Направление подготовки 09.04.04 Программная инженерия  
Уровень образования магистратура  
Отделение школы (НОЦ) Отделение информационных технологий  
Период выполнения весенний семестр 2019/2020 учебного года

Форма представления работы:

**Магистерская диссертация**

**КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН  
выполнения выпускной квалификационной работы**

Срок сдачи студентом выполненной работы:	10.06.2020
--	------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
10.03.2020	Аналитический обзор	25
14.04.2020	Объект и методы исследования	25
05.05.2020	Результаты	25
15.05.2020	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	15
01.06.2020	Социальная ответственность	10

**Составил преподаватель:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Аксёнов Сергей Владимирович	к.т.н.		

**СОГЛАСОВАНО:**

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Губин Евгений Иванович	к.ф.-м.н.		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА  
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И  
РЕСУРСОСБЕРЕЖЕНИЕ»**

Студенту:

Группа	ФИО
8ПМ8И	Войцеховский Алексей Алексеевич

Школа	ИШИТР	Отделение школы (НОЦ)	Информационных технологий
Уровень образования	Магистратура	Направление/специальность	Программная инженерия 09.04.04

**Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:**

1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих	Оклад руководителя - 33664 руб. Оклад инженера - 21760 руб. Стоимость материальных ресурсов определялась согласно прейскурантам компаний
2. Нормы и нормативы расходования ресурсов	Накладные расходы 16%; Районный коэффициент 30%. Норма амортизации 33%
3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования	Коэффициент отчислений на уплату во внебюджетные фонды 30 %.

**Перечень вопросов, подлежащих исследованию, проектированию и разработке:**

1. Оценка коммерческого и инновационного потенциала НТИ	Анализ перспективности технических решений посредством Quad-анализа SWOT-анализ. Оценка готовности научно-исследовательского проекта к коммерциализации
2. Разработка устава научно-технического проекта	Определение цели научно-исследовательского проекта, требований к проекту, описание заинтересованных стороны проекта, рабочей группы.
3. Планирование процесса управления НТИ: структура и график проведения, бюджет, риски и организация закупок	Планирование этапов работы, определение календарного графика проведения исследования Определение рисков научно-исследовательского проекта, оценка вероятности риска и потерь Расчет бюджета затрат на проведение исследования
4. Определение ресурсной, финансовой, экономической эффективности	Описание потенциального эффекта научно-исследовательского проекта.

**Перечень графического материала (с точным указанием обязательных чертежей):**

1. Оценочная карта технологии QuaD
2. Матрица SWOT
3. Диаграмма Ганта
4. Бюджет затрат
5. Реестр рисков

**Дата выдачи задания для раздела по линейному графику**

**Задание выдал консультант:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Меньшикова Екатерина Валентиновна	к.ф.н.		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8ПМ8И	Войцеховский Алексей Алексеевич		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА  
«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»**

Студенту:

<b>Группа</b>	<b>ФИО</b>
8ПМ8И	Войцеховский Алексей Алексеевич

<b>Школа</b>	<b>ИШИТР</b>	<b>Отделение (НОЦ)</b>	<b>Информационных технологий</b>
Уровень образования	Магистратура	Направление/специальность	Программная инженерия 09.04.04

Тема ВКР:

<b>Способ сегментации лёгких на снимках КТ с использованием методов глубокого обучения</b>	
<b>Исходные данные к разделу «Социальная ответственность»:</b>	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	Объектом исследования является алгоритм сегментации лёгких на снимках компьютерной томографии. Алгоритм выполняется на персональном компьютере. Рабочим местом является место за персональным компьютером.
<b>Перечень вопросов, подлежащих исследованию, проектированию и разработке:</b>	
<b>1. Правовые и организационные вопросы обеспечения безопасности:</b> – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны.	Требования к организации рабочего места: – Трудовой кодекс; – Федеральный закон N 426-ФЗ "О специальной оценке условий труда"; – Приказ Минтруда N 33н "Об утверждении Методики проведения специальной оценки условий труда"; – Приказ Минздравсоцразвития N 302н "Об утверждении перечней вредных и (или) опасных производственных факторов и работ, при выполнении которых проводятся обязательные предварительные и периодические медицинские осмотры".
<b>2. Производственная безопасность:</b> 2.1. Анализ выявленных вредных и опасных факторов 2.2. Обоснование мероприятий по снижению воздействия	Анализ выявленных вредных факторов: – Шум – Микроклимат – Освещение – Напряженная зрительная работа – Монотонность труда Анализ выявленных опасных факторов: – Электрический ток (источником является ПК и лабораторное электрооборудование) – Статическое электричество – Короткое замыкание
<b>3. Экологическая безопасность:</b>	Анализ воздействия на литосферу и атмосферу (образование отходов при поломке ПК, перегорании

	люминесцентных ламп)
<b>4. Безопасность в чрезвычайных ситуациях:</b>	Чрезвычайной ситуацией, которая может возникнуть на рабочем месте, является пожар.

Дата выдачи задания для раздела по линейному графику	
--	--

**Задание выдал консультант:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Горбенко Михаил Владимирович	к.т.н.		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8ПМ8И	Войцеховский Алексей Алексеевич		

## Планируемые результаты обучения по направлению

### 09.04.04 «Программная инженерия»

Код результата	Результат обучения (выпускник должен быть готов)
<b>Общие по направлению подготовки 09.04.04 «Программная инженерия»</b>	
P1	Проводить научные исследования, связанные с объектами профессиональной деятельности
P2	Разрабатывать новые и улучшать существующие методы и алгоритмы обработки данных в информационно-вычислительных системах
P3	Составлять отчеты о проведенной научно-исследовательской работе и публиковать научные результаты
P4	Проектировать системы с параллельной обработкой данных и высокопроизводительные системы
P5	Осуществлять программную реализацию информационно-вычислительных систем, в том числе распределенных
P6	Осуществлять программную реализацию систем с параллельной обработкой данных и высокопроизводительных систем
P7	Организовывать промышленное тестирование создаваемого программного обеспечения
<b>Профиль «Технологии больших данных» / «Big data solutions»</b>	
P8	Исследовать и анализировать большие данные, создавать их модели и интерпретировать структуры данных в таких моделях
P9	Понимать принципы создания, хранения, управления, передачи и анализа больших данных с использованием новейших технологий, инструментов и систем обработки данных в высокопроизводительных сетях
P10	Применять теорию распределенной системы управления базами данных к традиционным распределенным системам реляционных баз данных, облачным базам данных, крупномасштабным системам машинного обучения и хранилищам данных

## Реферат

Работа содержит пояснительную записку на 106 листах, содержит 23 рисунка, 28 таблиц, 2 приложения, 42 источника.

Ключевые слова: глубокое обучение, компьютерное зрение, компьютерная томография, анализ медицинских изображений, семантическая сегментация.

Цель данной работы – разработка способа сегментации легких на снимках КТ с использованием методов глубокого обучения для повышения эффективности работы врача-радиолога для описания состояния легких по данным КТ. Разработанное программное обеспечение позволяет сегментировать область легкого на срезе с помощью технологий глубокого обучения. Результаты работы планируется использовать подразделением лучевой диагностики СибГМУ.

## Оглавление

Реферат .....	10
Оглавление.....	11
Введение.....	14
1 Обзор литературы .....	15
1.1 Глубокое обучени .....	15
1.2 Переобучение нейронных сетей и методы предотвращения переобучения .....	17
1.3 Свёрточные нейронные сети.....	19
1.3 Архитектуры для семантической сегментации .....	23
1.4 Архитектуры энкодеров .....	24
1.3.1 Архитектура ResNet.....	25
1.3.2 Архитектура MobileNetV2 .....	26
1.3.3 Архитектура EfficientNet.....	27
1.4 Предварительное обучение.....	28
1.5 Аугментация данных .....	28
2 Объект и методы исследования.....	29
2.1 Описание используемых программных и аппаратных средств .....	29
2.1 Данные для обучения и тестирования .....	30
2.2 Аугментация обучающей выборки .....	31
2.3 Обучение моделей.....	33
3 Результаты проведенного исследования .....	35
4 Финансовый менеджмент, ресурсоэффективность и энергосбережение .....	40
4.1 Предпроектный анализ.....	40
4.2 Технология QuaD .....	41
4.3 SWOT-анализ.....	42
4.4 Оценка готовности научно-исследовательского проекта к коммерциализации .....	47
4.5 Инициация научно-исследовательского проекта .....	49
4.6 Планирование научно-исследовательских работ .....	50
4.6.1 Организационная структура научно-исследовательского проекта.....	50
4.6.2 Структура работ в рамках научного исследования.....	51
4.6.3 Определение трудоемкости выполнения работ .....	52

4.6.4	Разработка графика проведения научного исследования.....	56
4.7	Бюджет научно-технического исследования .....	58
4.7.1	Расчет материальных затрат .....	58
4.7.2	Расчет затрат на специальное оборудование для научных работ.....	59
4.7.3	Расчет затрат на амортизацию оборудования.....	59
4.7.4	Основная заработная плата исполнителям работы .....	60
4.7.5	Дополнительная заработная плата .....	62
4.7.6	Отчисления во внебюджетные фонды.....	62
4.7.7	Накладные расходы .....	63
4.7.8	Контрагентные расходы .....	64
4.7.9	Формирование бюджета затрат научно-исследовательского проекта .....	64
4.8	Риски научно-исследовательского проекта .....	65
4.9	Выводы по разделу финансовый менеджмент.....	66
5	Социальная ответственность .....	67
5.1	Введение .....	67
5.2	Правовые и организационные вопросы обеспечения безопасности.....	67
5.3	Профессиональная социальная безопасность.....	69
5.3.1	Отклонение показателей микроклимата.....	70
5.3.2	Превышение уровня шума .....	71
5.3.3	Освещение .....	72
5.3.4	Психофизиологические факторы при работе с компьютером ...	76
5.3.5	Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека.....	77
5.4	Экологическая безопасность .....	78
5.5	Безопасность в чрезвычайных ситуациях .....	79
5.6	Выводы по разделу .....	81
	Заключение .....	82
	Список публикаций и научных достижений.....	83
	Список используемых источников.....	84
	Приложение А .....	88
1.1	Deep learning .....	89
1.2	Neural networks overfitting and methods for preventing overfitting...	91
1.3	Convolutional neural networks .....	92

1.4 Semantic segmentation architectures .....	95
1.5 Encoders architectures .....	97
1.5.1 ResNet .....	97
Figure 4. Residual block in ResNet .....	98
1.5.2 MobileNetV2 .....	98
1.5.3 EfficientNet .....	99
1.6 Transfer learning .....	99
1.7 Data augmentation .....	100
References .....	101
Приложение Б. Исходный код процесса обучения и проверки моделей .....	103

## **Введение**

Современное компьютерное зрение является динамично развивающейся областью информационных технологий. Во многом это связано с широким распространением методов глубокого обучения, способных решать множество задач, в том числе связанных с анализом медицинских изображений.

Компьютерное зрение активно развивается в качестве инструмента автоматизации диагностики заболеваний. В том числе производятся попытки автоматического поиска патологий лёгких и других органов на снимках компьютерной томографии и других методов диагностики. Для того чтобы правильно найти заболевание на этих изображениях, имеет смысл сначала выделить непосредственно исследуемый орган. Поэтому в данной работе рассматривается использование методов глубокого обучения для сегментации лёгких на снимках КТ.

## **1 Обзор литературы**

### **1.1 Глубокое обучени**

Глубокое обучение является совокупностью методов машинного обучения, основанных на обучении представлению. Глубокое обучение в настоящее время является state of the art подходом в решении задач компьютерного зрения, обработки естественной речи, распознавая речи и т.д. Успехи глубокого обучения связаны с развитием вычислительных мощностей, в том числе графических и специализированных тензорных процессоров [1].

Несмотря на то, что глубокое обучение получило широкое распространение только в недавнее время, попытки обучать алгоритмы, основанные на многослойных нейронных сетях, осуществлялись с 60-х годов 20 века [2]. В 1980 году Кунихико Фукушима представил модель Неокогнитрон, предназначенную для распознавания визуальных образов [3].

В 1989 году Ян ЛеКун и остальные из AT&T Bell Laboratories предложили использовать алгоритм обратного распространения ошибки для обучения глубоких свёрточных сетей. В рамках работы они обучили свёрточную нейронную сеть, которая могла распознавать почтовые ZIP-коды. Обучающая выборка включала в себя 9298 изображений цифр, подготовленных на основе написанных вручную ZIP-кодов. Время обучения алгоритма составляло 3 дня. Финальная модель была реализована на процессоре цифровой обработки сигналов и могла обрабатывать 10 цифр в секунды [4].

В последующие годы было реализовано множество работ так или иначе развивающих методику применения глубокого обучения. Однако прорыв произошел только в 2012 году, когда Алекс Крижевский, Илья Суцкевер и Джеффри Хинтон из Университета Торонто выиграли соревнование по классификации изображений ImageNet и опубликовали статью [5]. До этого в ImageNet побеждали классические алгоритмы компьютерного зрения, которые использовали подобранные вручную признаки. При обучении модели AlexNet

использовали регуляризацию с помощью метода “dropout”, аугментацию обучающей выборки другие техники глубокого обучения. Одной из особенностей было использование двух параллельных графических процессора NVidia GTX 580. Именно использование графических процессоров для распараллеливания вычислений нейронных сетей и положило начало широкому применению глубокого обучения, как в академической среде, так и в коммерческой сфере.

Также стоит отметить еще один фактор, позволивший глубоким моделям достичь текущих успехов – наличие большого количества данных. На рисунке показано, как увеличивались публичные наборы данных со временем [6]

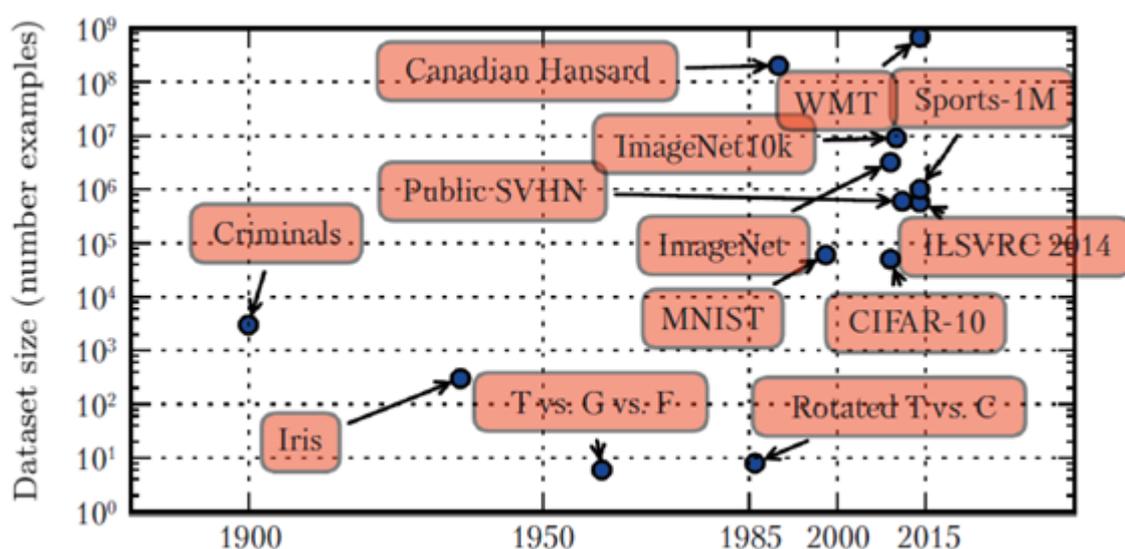


Рисунок 1 – Увеличение размера набора данных со временем

В начале 1900-х годов статистики изучали наборы данных, содержащие от сотен до тысяч вручную подготовленных измерений. В период между 1950-ми и 1980-ми пионеры компьютерного зрения зачастую работали с небольшими синтетическими наборами данных, например, растровыми изображениями букв низкого разрешения, специально спроектированными так, чтобы снизить стоимость вычислений и продемонстрировать, что нейронные сети можно обучить функциям специального вида. В 1980-е и 1990-е машинное обучение стало в большей степени статистическим, а наборы данных уже насчитывали десятки тысяч примеров, как, например, набор отсканированных рукописных

цифр MNIST. В первом десятилетии XXI века продолжали создавать более изощренные наборы данных того же размера, например CIFAR-10. В конце этого периода и в первой половине 2010-х появление больших наборов данных, содержащих от сотен тысяч до десятков миллионов примеров, полностью изменило представление о возможностях глубокого обучения. К таким наборам относится общедоступный набор номеров домов Street View House Numbers, различные варианты набора ImageNet и набор Sports-1M. В верхней части диаграммы мы видим, что наборы переведенных предложений, например набор, построенный IBM по официальным отчетам о заседаниях канадского парламента, и набор WMT 2014 переводов с английского на французский, по размеру намного превосходят большинство остальных наборов [6].

## **1.2 Переобучение нейронных сетей и методы предотвращения переобучения**

Одной из проблем, с которой сталкиваются методы глубокого обучения, является переобучение. В машинном обучении под переобучением понимают явление, когда обученная модель плохо работает на примерах, которые отсутствовали в обучающем наборе данных. Переобучение происходит, когда модель обучается на скрытых случайных закономерностях в обучающих данных, но отсутствующих в генеральной совокупности. Переобучение характеризует обобщающую способность модели.

Для решения проблемы переобучения глубоких нейронных сетей применяют следующие техники регуляризации:

1. L1- и L2-регуляризация весов
2. Ранняя остановка обучения
3. Метод “dropout”

Смысл первой техники заключается в ограничение абсолютных значений весов нейронной сети. Для этого к функции потерь применяется один из регуляризаторов:

1. L1-регуляризатор – сумма квадратов весов  $L1 = \lambda \sum_{\omega} \omega^2$

2. L2-регуляризатор – сумма модулей весов  $L2 = \lambda \sum_{\omega} |\omega|$

Вторая техника является довольно простой. Необходимо отложить часть обучающей выборки в валидационную выборку. Затем на каждой эпохе обучения вычислять ошибку на этой выборке – если ошибка начинает расти после определенного минимума, значит, нейронная сеть начала переобучаться и нужно остановить процесс обучения. Есть данные, что данная техника в некоторой степени близка к L2-регуляризации [7]. Также анализ ошибки на валидационной выборке позволяет оценивать обобщающую способность модели.

Третий метод также нашел широкое применение для предупреждения переобучения нейронных сетей, т.к. является более эффективным, чем первый, и часто применяется в первую очередь. Идея метода “dropout” заключается в следующем – для каждого нейрона кроме нейронов выходного слоя определяется вероятность  $p$ . С этой вероятностью нейрон отключается. Алгоритм обучения становится таким: на каждом сэмпле  $x$  из обучающей выборки вычисляется, будет ли активен нейрон, и в зависимости от результата нейрон либо остается активным, либо отключается и его выход устанавливается равным нулю. Ноль на выходе приводит к тому, что нейрон отсутствует в вычислительном графе. Прямой проход, и обратное распространение градиента не идут дальше неактивного нейрона [8]. Эксперименты [9], и практика обучения нейронных сетей показывают, что метод “dropout” действительно дает очень серьезные улучшения в качестве обученной модели в самых разных приложениях.

### 1.3 Свёрточные нейронные сети

Обычно для решения задач компьютерного зрения применяют свёрточные нейронные сети. Первой реализованной свёрточной нейронной сетью считается Неокогнитрон, хотя этот алгоритм не использовал методы обучения с учителем [3]. Принято считать, что свёрточные нейронные сети в некоторой степени имитируют работу зрительной коры головного мозга [10]. Зрительная кора делится на области названные visual area one (V1) и так далее до V7. Эти области имеют разную структуру, физиологию и функции [10]. Современная нейрофизиология выделяет следующие функции этих зон:

1. V1 выделяет локальные признаки небольших областей изображений с сетчатки;
2. V2 также выделяет локальные признаки и обобщает их, а также отвечает за бинокулярное зрение;
3. V3 распознает цвета и текстуры объектов, а также сегментирует их;
4. V4 распознает геометрические формы и отвечает за фокусирование на конкретном объекте;
5. V5 распознает направление и скорость движения объектов;
6. V6 обобщают картину целиком;
7. V7 распознает сложные объекты, в том числе человеческие лица.

Глубокие свёрточные нейронные сети имеют схожую со зрительной корой структуру: первые слои выделяют простые признаки – границы и цвета. Последующие слои на основе этих признаков выделяют более сложные признаки, например, формы. Последние слои отвечают за распознавание объектов [10].

Главная идея свёрточных нейронных сетей заключается в том, обработка части изображения должна происходить независимо от местоположения этой части. Реализуется эта идея явно – мы покрываем входное изображение небольшими окнами (например, 5x5 пикселей) и выделяем признаки в каждом

таком окне небольшой нейронной сетью. При этом признаки выделяются в каждом окне одни и те же, значит, будет одна нейронная сеть. У нее будет  $5 \times 5 = 25$  входов. Из каждого изображения получится много разных входов для этой нейронной сети. После этого выход можно будет снова представить в виде изображения, заменив окна  $5 \times 5$  на их центральные пиксели. На этом изображении можно будет применить второй свёрточный слой с уже другой нейронной сетью. В каждом свёрточном слое будет немного свободных параметров по сравнению с полносвязными нейронными сетями.

Обычно исходное изображение представляет собой три матрицы (по цветовым каналам RGB) – красный, зеленый и голубой. Такие матрицы называют каналами. Такие же матрицы будут получаться после каждого свёрточного слоя. В них присутствует пространственная структура, соответствующая исходной картинке. Каналов становится больше. Значения каждого признака, которые мы выделили из окон в исходном изображении, теперь будут представлять собой целую матрицу. Такие матрицы принято называть картами признаков.

Далее разберемся, как устроены свёрточные слои. Операция свёртки это линейное преобразование входных данных особого вида. Пусть  $x^l$  — карта признаков в слое  $l$ . Тогда результат двумерной свертки с ядром размера  $2d+1$  и матрицей весов  $W$  размера  $(2d + 1) \times (2d + 1)$  на следующем слое будет равен:

$$y_{i,j}^l = \sum_{-d \leq a, b \leq d} W_{a,b} x_{i+a, j+b}^l$$

где  $y_{i,j}^l$  – результат свертки на уровне  $l$ , а  $x_{i,j}^l$  – вход свёртки, он же выход предыдущего слоя. Для получения результата на следующем слое, мы применяем свёртку к квадратному окну с предыдущего слоя, то есть скалярно умножаем пиксели окна на вектор свёртки.

Операция свёртки обладает следующими свойствами:

8. Свёртка сохраняет структуру входного изображения, так как применяется к каждому участку входных данных в отдельности;

9. Операция свертки обладает свойством разреженности, так как значение каждого нейрона очередного слоя зависит только от небольшой доли входных нейронов;

10. Свёртка многократно переиспользует одни и те же веса, так как они повторно применяются к различным участкам входа.

После операции свёртки можно, применить ту или иную нелинейную функцию  $h$ : она будет просто применяться к каждому элементу полученного тензора по отдельности. В классическом сверточном слое, кроме линейной свертки и следующей за ней функцией активации, есть и еще одна операция: пулинг.

В свёрточных сетях важнее наличие или отсутствие какого-либо признака, чем его точные координаты. Например, при распознавании лиц важно понять, есть ли на изображении лицо и кому оно принадлежит, а не узнать, где оно расположено на изображении. Поэтому мы можем позволить себе обобщить выделяемые признаки, при этом потеряв часть информации об их расположении. К тому же это дает нам возможность сократить размерность [6].

В основном в качестве операции пулинга применяется операция взятия максимума (max-pooling). Формула для расчёта пулинга:

$$x_{i,j}^{l+1} = \max_{-d \leq a \leq d, -d \leq b \leq d} z_{i+a, j+b}^l$$

Пример работы операции макс-пулинг представлен на рисунке 2.

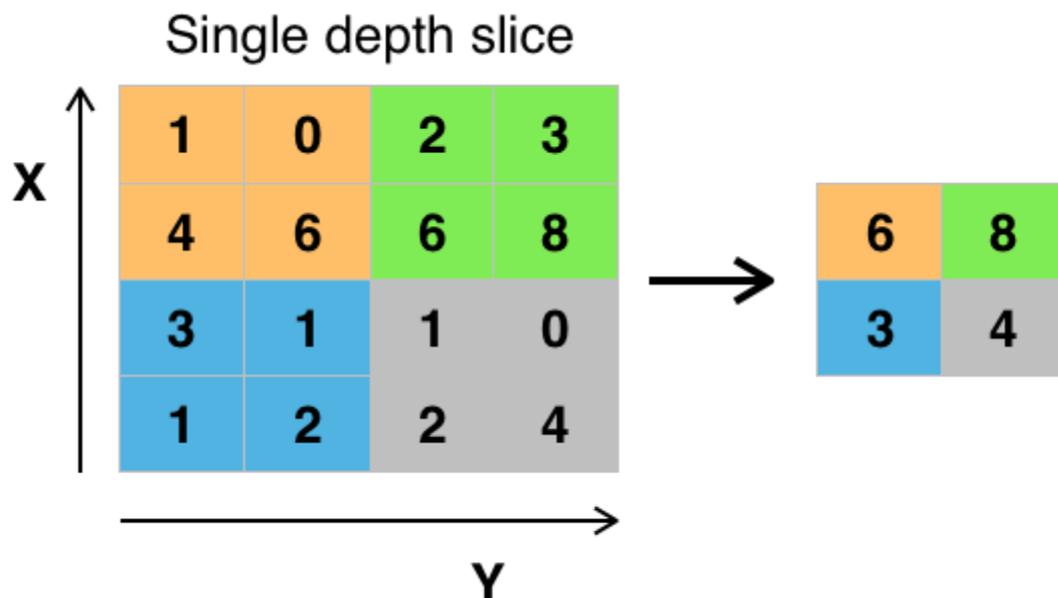


Рисунок 2 – Макс-пулинг

Другим важным элементом свёрточных нейронных сетей является функция активации, о которой было сказано ранее. Функция активации после слоя свёртки является источником нелинейности, которая дает нам возможность обучать нейронные сети. Функция активации должна отвечать следующим критериям:

1. Быть монотонно неубывающей
2. Быть ограниченной, т.е. стремиться к нулю при  $x \rightarrow -\infty$
3. Быть дифференцируемой

Существует множество функция активации: логистическая сигмоида, гиперболический тангенс и другие. Но наиболее популярной функцией активации для свёрточных нейронных сетей является rectified linear units (ReLU) [10]. Формула ReLU имеет следующий вид:

$$ReLU(x) = \begin{cases} 0, & \text{если } x < 0, \\ x, & \text{если } x \geq 0. \end{cases}$$

Функция ReLU более эффективна, чем логистическая регрессия или гиперболический тангенс, т.к. вычисление её производной не требует сложных расчётов: если  $x < 0$ , тогда производная равна нулю, если  $x \geq 0$ , тогда производная равна единице. Это также позволяет уменьшить расходы на

вычислительные ресурсы. Кроме классической реализации ReLU существуют её модификации, позволяющие получить лучший результат в некоторых задачах [6].

### **1.3 Архитектуры для семантической сегментации**

В данной работе решается задача семантической сегментации. Семантическая сегментация в компьютерном зрении подразумевает присвоение каждому пикселю изображения метки определенного класса. Например, в случае задачи сегментации лёгких каждый пиксель необходимо отнести к одному из двух классов – лёгкое или фон.

В качестве модели для семантической сегментации обычно используется архитектура U-net [11]. U-net была представлена Олафом Ронненбергом и остальными в 2015 году. На тот момент имеющиеся архитектуры для сегментации изображений требовали обучающие выборки, состоящие из тысяч изображений. Для решения этой проблемы при решении задач обработки биомедицинских изображений и была предложена эта архитектура. Она позволила достичь хороших результатов даже на небольших выборках данных. Кроме того, U-net является довольно лёгкой архитектурой и не требует существенных вычислительных ресурсов. Также она является универсальной и позволяет решать задачи сегментации, как биомедицинских изображений, так и изображений из других областей, например, задачу сегментации объектов на спутниковых снимках [12].

Рассмотрим архитектуру U-net подробнее. Схема U-net представлена на рисунке 4 [11].

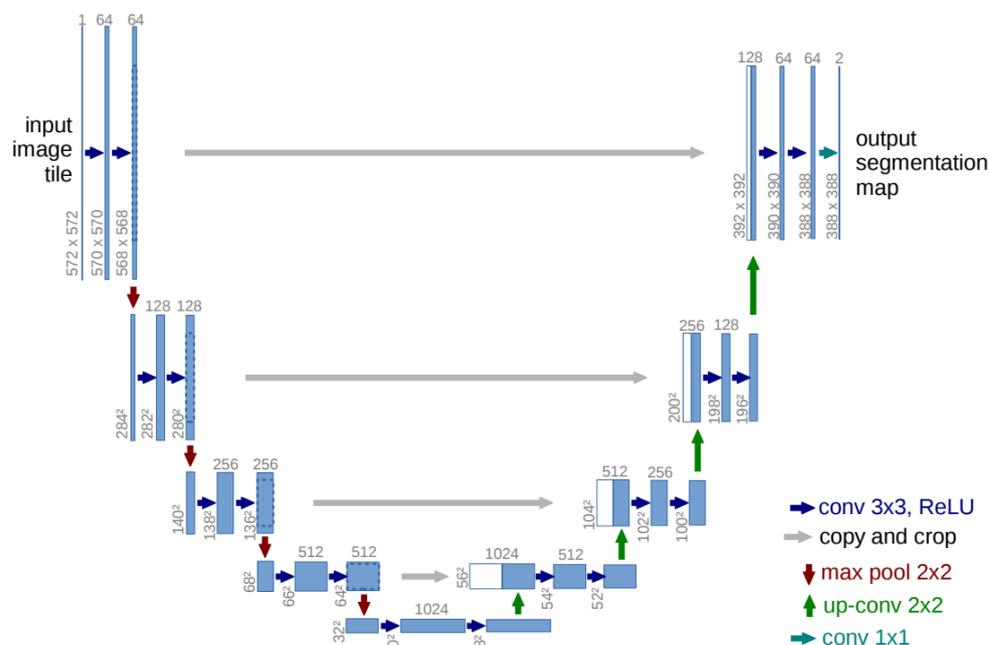


Рисунок 4 – Общая схема архитектуры U-net

Как видно из рисунка, U-net имеет сжимающую часть слева и расширяющую часть справа. Сжимающая часть является типичной архитектурой свёрточных нейронных сетей. Она содержит в себя блоки свёртки, функции активации ReLU и объединения максимумом. Расширяющая часть состоит также из блоков свёртки и функции активации ReLU, но вместо блоков объединения применяются обратные им блоки. Также карты признаков перед каждым блоком объединения передаются на соответствующие слои расширяющей части и конкатенируются. При этом края обрезаются, т.к. свёртка приводит к потере пикселей на краях изображения. Последний слой представляет собой свёртку 1x1 и предназначен для отображения 64 карт признаков в выходное изображение с необходимым числом классов. В данной работе требуется 2 класса – лёгкие и фон.

## 1.4 Архитектуры энкодеров

Классический U-net, представленный ранее содержит в себе 23 слоя. Тем не менее, для решения многих задач находит широкое применение измененная

архитектура U-net. В ней сжимающая часть заменяется какой-либо распространенной свёрточной архитектурой, обычно предобученной на наборе данных из близкого домена или же на наборе с большим числом классов и примеров, например, ImageNet [13]. Часто это позволяет увеличить точность сегментации за счёт увеличения количества слоёв и применения других техник повышения качества работы модели [12]. В данной работе в качестве энкодеров для U-net были используются следующие архитектуры:

1. ResNet-34 [14]
2. MobileNetV2 [16]
3. EfficientNet-B0 [17]

Далее рассмотрим их подробнее.

### **1.3.1 Архитектура ResNet**

Семейство архитектур свёрточных нейронных сетей ResNet было предложено в 2015 году группой исследователей из компании Microsoft [14]. Чем глубже нейронная сеть, тем сложнее её обучать. Поэтому Кайминг Хе и остальные предложили подход, названный остаточными нейронными сетями (англ. residual network – ResNet). Основной смысл этого подхода в добавлении путей обхода групп слоёв нейронной сети, образуя остаточный блок, как показано на рисунке 5 [14]. Использование такого блока позволяет достичь глубины нейронной сети в 152 слоя, что в 8 раз больше, чем у глубоких сетей VGG [15]. При этом сети ResNet сохраняют меньшую вычислительную сложность [14]. В результате такой подход позволяет сравнительно просто увеличивать точность нейронной сети при увеличении числа слоёв.

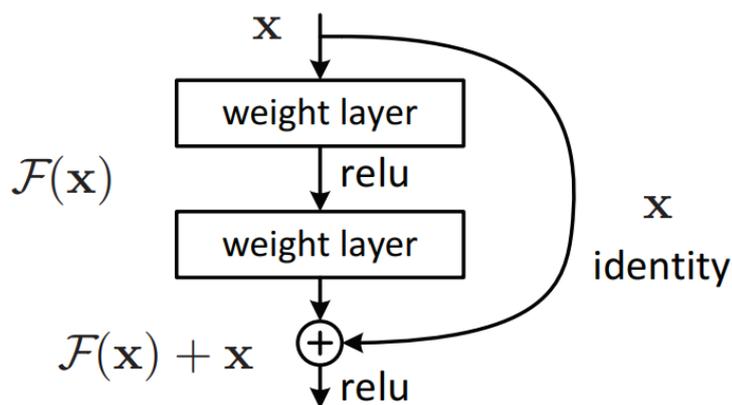


Рисунок 5 – Остаточный блок в сетях ResNet

### 1.3.2 Архитектура MobileNetV2

Архитектура MobileNetV2 была предложена в 2019 году Марком Сэндлером и остальными из компании Google [16]. Целью этой работы было достижение нового уровня качества на мобильных платформах, например, смартфонах. Задача усложняется за счёт требований к габаритам и энергоэффективности мобильных процессоров. Данная архитектура основывается на ранее рассмотренной ResNet. Особенностью данной архитектуры является использование свёрточных блоков с бутылочным горлышком, не содержащим нелинейностей. На рисунках 6 и 7 представлены обычный блок свёртки и блок свёртки с бутылочным горлышком (заштрихованный слой не содержит нелинейностей) [16].

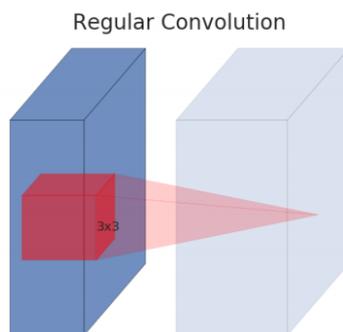


Рисунок 6 – Обычный блок свёртки

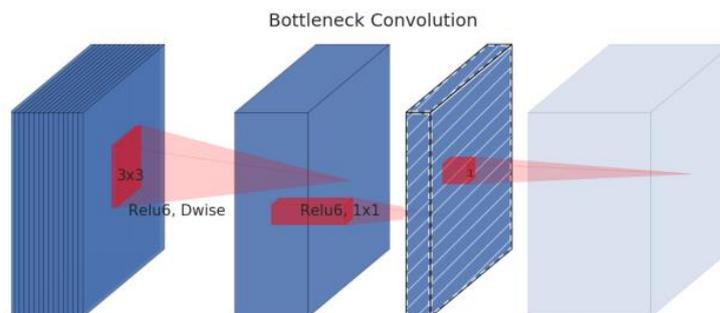


Рисунок 7 – Блок свёртки с бутылочным горлышком

Второй особенностью является использования вместо обычных остаточных блоков обратных остаточных блоков. Их отличие заключается в том, что обход соединяет блоки с бутылочными горлышками. В целом такая архитектура позволяет найти компромисс между точностью и производительностью, что может быть полезно, если модель необходимо использовать на слабых персональных компьютерах или мобильных устройствах.

### 1.3.3 Архитектура EfficientNet

Семейство архитектур EfficientNet было предложено в 2019 году Минсин Таном и Квоком Ле из компании Google [17]. Авторы использовали подход поиска нейронных архитектур (англ. Neural Architecture Search или NAS) для поиска новых архитектур. С помощью данного метода, варьируя глубину и ширину нейронной сети, и разрешение изображений, они получили новое семейство нейронных сетей EfficientNet. Новые нейронные сети показали более высокие результаты, чем существующие архитектуры, при меньшем размере и вычислительной сложности [17].

## **1.4 Предварительное обучение**

Как было сказано ранее, предварительное обучение нейронных сетей широко применяется при решении задач сегментации. Это обусловлено улучшением процесса сходимости нейронных сетей по сравнению со случайной инициализацией весов [12]. Наиболее распространенным набором данных для предварительного обучения служит ImageNet [13]. Этот выбор не случаен, т.к. ImageNet является крупнейшей базой размеченных изображений и содержит более 1000 классов, что позволяет подготовить нейронную сеть к большинству доменов.

## **1.5 Аугментация данных**

Аугментация обучающей выборки является распространенной практикой в машинном обучении. Под аугментацией подразумевается расширение обучающей выборки за счёт создания новых примеров на основе имеющихся с помощью различных искажений. Для изображений как правило применяют сдвиги, повороты, масштабирование, искажения, смещения цветовых каналов и др. Например, при работе над классической архитектурой U-net, авторы применяли различные аугментации для повышения качества работы модели [11]. Другим примером использования аугментации обучающей выборки является статья Владимира Игловиков о применении модифицированной U-net для сегментации изображений на спутниковых снимках, в которой рассматривается улучшение процесса обучения модели за счёт аугментации данных [12].

## **2 Объект и методы исследования**

### **2.1 Описание используемых программных и аппаратных средств**

Для обучения моделей сегментации на основе нейронных сетей был выбран фреймворк с открытым исходным кодом Tensorflow, разрабатываемый компанией Google [18]. Он имеет интерфейс для работы с языком программирования Python, что позволяет быстро прототипировать модели на основе глубокого обучения. Бэкенд фреймворка написан на C++, что дает высокую производительность при обучении и использовании глубоких нейронных сетей [18]. Кроме того, с недавнего времени Tensorflow включает в себя высокоуровневый модуль Keras, позволяющий создавать модели глубокого обучения из готовых слоев и других операций.

Для использования предобученных модели U-net с модифицированными энкодерами была выбрана библиотека Segmentation Models для фреймворка Keras, разработанная Павлом Якубовским [19]. Данная библиотека содержит распространенные архитектуры и функции потерь для решения задач сегментации объектов.

Для обучения моделей использовался сервис Paperspace, предоставляющий инфраструктуру с GPU в аренду. Для энтузиастов, исследователей и студентов предоставляется бесплатный доступ, если имеются свободные GPU. В качестве GPU выступают графические ускорители Nvidia Quadro P5000. Работа осуществляется в среде на основе Jupyter Notebook, что освобождает от необходимости самостоятельно администрировать виртуальные машины или контейнеры. Интерфейс сервиса представлен на рисунках 8 и 9. Через 6 часов после начала сессии, данные сохраняются и виртуальная машина выключается. Для дальнейшей работы можно заново запустить сессию, и если есть свободные GPU, то использовать их.

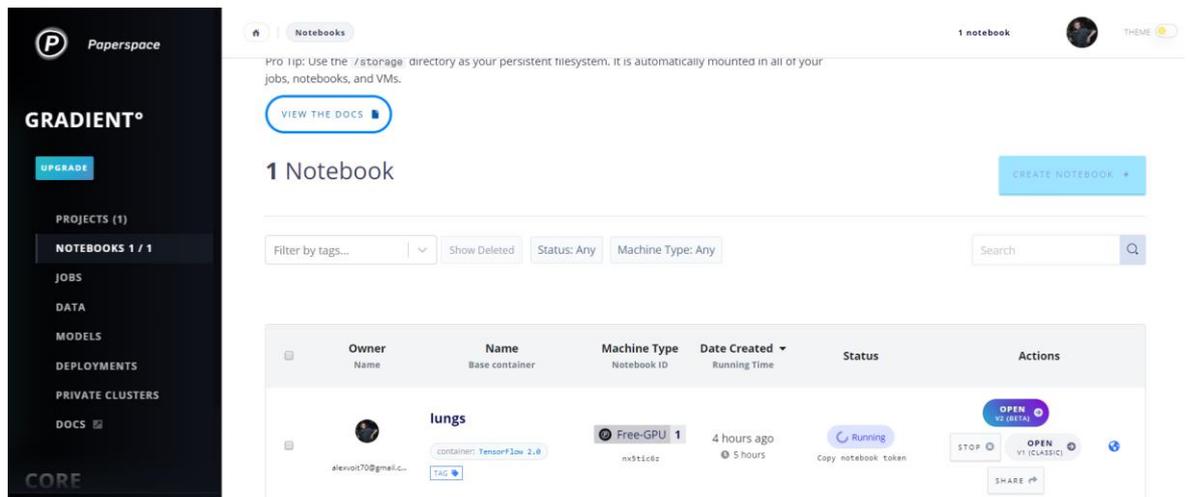


Рисунок 8 – Интерфейс выбора ноутбука

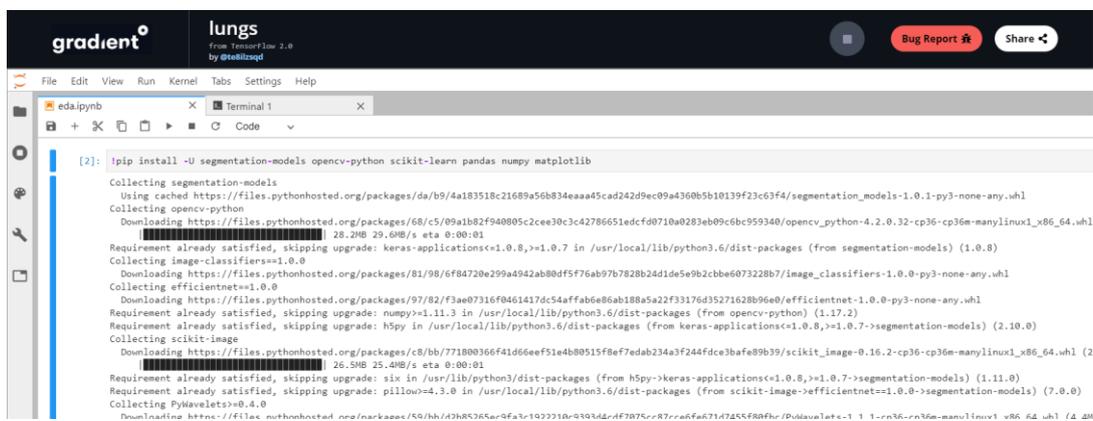


Рисунок 9 – Интерфейс Jupyter Lab

## 2.1 Данные для обучения и тестирования

Для обучения и тестирования модели глубокого обучения для сегментации легких на снимках КТ был использован датасет Finding and Measuring Lungs in CT Data с платформы Kaggle, представленный Кевином Мейдером [20]. Данный набор содержит 267 снимков и масок легких. Данные представлены в формате изображений tiff. Пример снимка и маски легких из датасета представлен на рисунке 10.

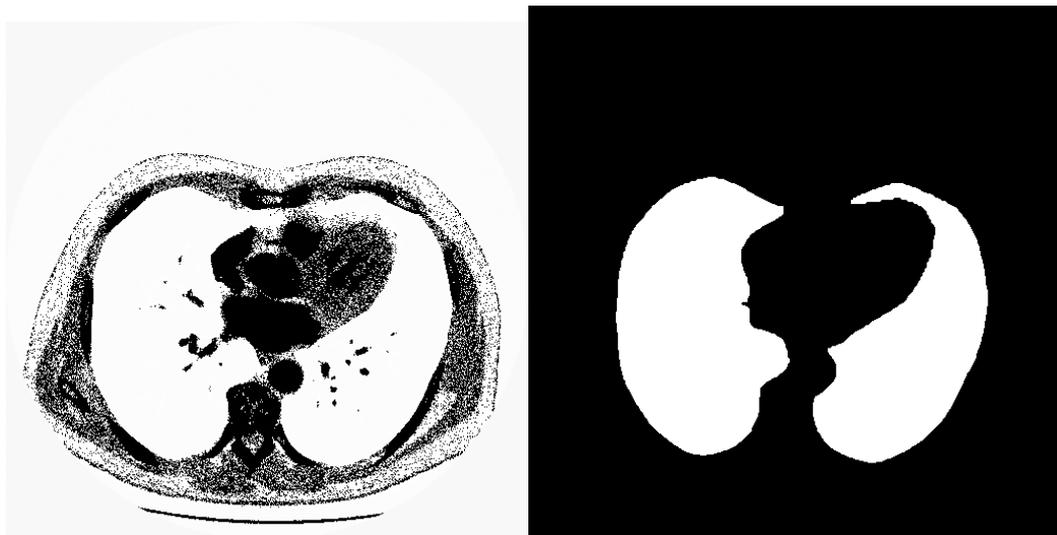


Рисунок 10 – Снимок КТ и соответствующая маска

## 2.2 Аугментация обучающей выборки

Как было сказано ранее, аугментация обучающей выборки является распространенной выборкой для улучшения процесса обучения моделей, в том числе глубоких свёрточных сетей. Для увеличения размера обучающей выборке были использованы следующие аугментации:

1. Поворот на угол до 5 градусов
2. Сдвиги по высоте и ширине до 5%
3. Увеличение до 20%
4. Горизонтальные повороты

Примеры аугментированных изображений и их маски представлены на рисунках 11-13. Можно заметить, что изображения иногда искажены достаточно сильно, но свёрточные нейронные сети инвариантны к свигам объектов на изображении [4], поэтому такие аугментации не ухудшают обобщающую способность используемой модели.

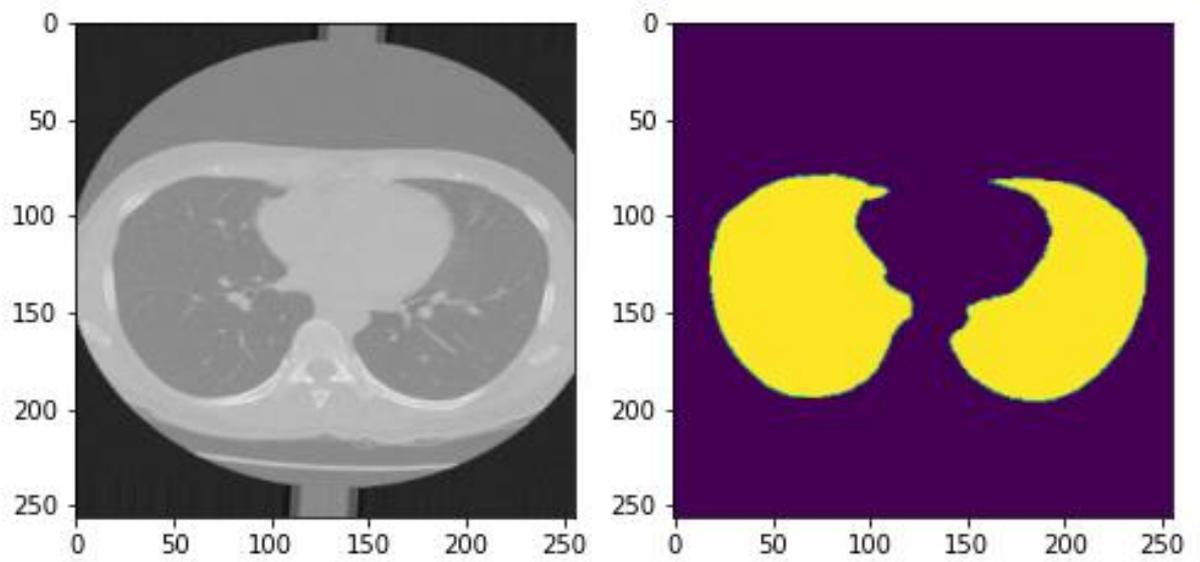


Рисунок 11 – Пример аугментированного изображения и соответствующая маска №1

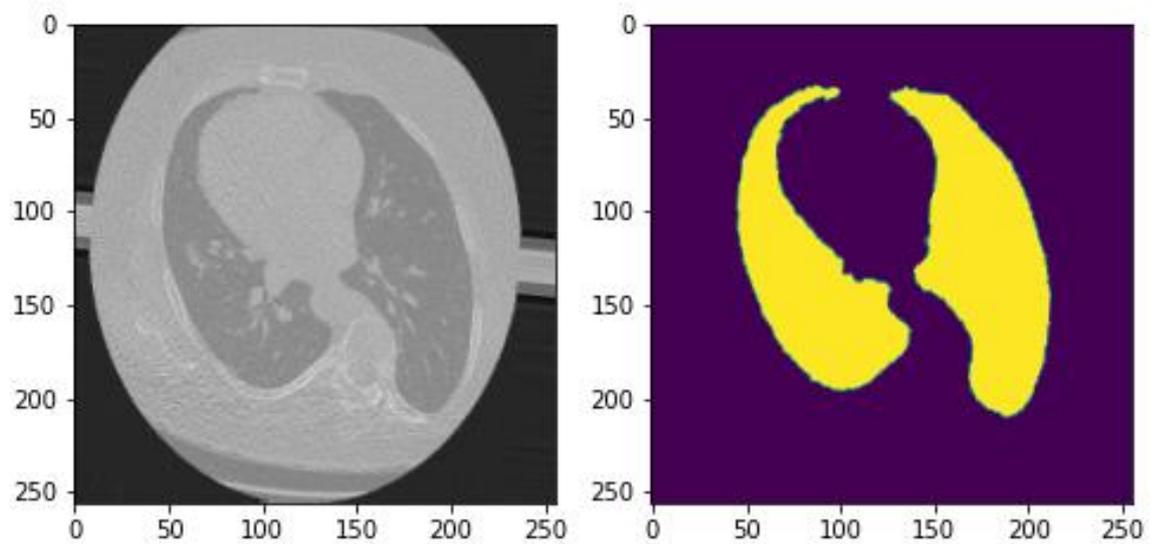


Рисунок 12 – Пример аугментированного изображения и соответствующая маска №2

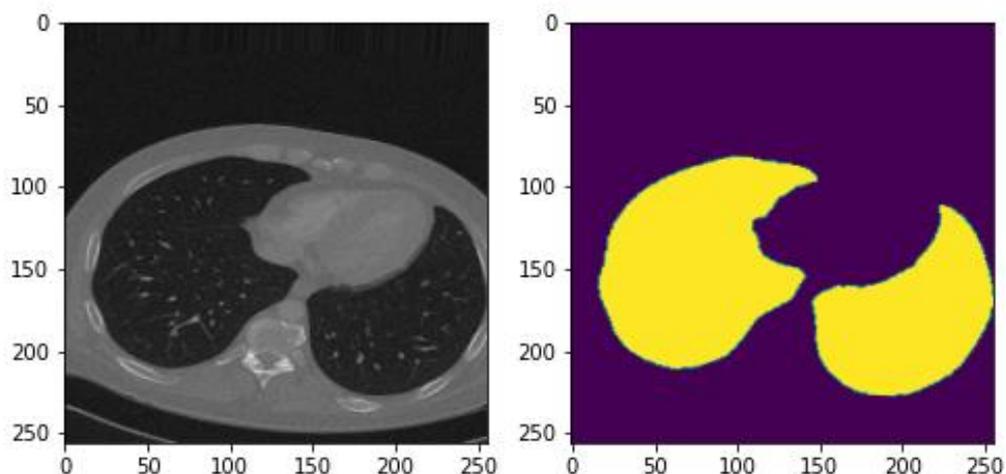


Рисунок 13 – Пример аугментированного изображения и соответствующая маска №3

### 2.3 Обучение моделей

Данные были разбиты на обучающую, валидационную и тестовую выборки в соотношении 50:25:25. Обычно соотношения разбиения подбираются эмпирически.

Для оптимизации весов модели использовалась функция потерь Жаккарда, а для расчета качества метрика intersection over union (IoU) (рисунок 14). Исходный код процесса обучения и проверки моделей представлен в приложении Б.

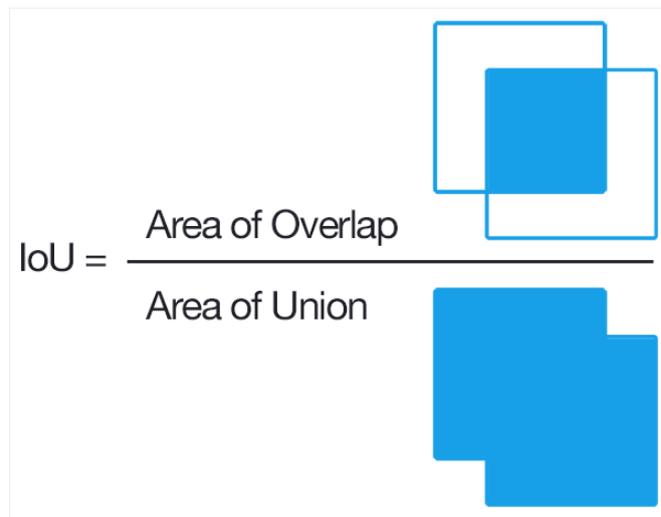


Рисунок 14 – Схема определения IoU

### 3 Результаты проведенного исследования

Результаты обучения трёх моделей представлены в таблице 1.

Таблица 1 – Результаты обучения

<b>Энкодер</b>	<b>Время обучения, эпох</b>	<b>IoU на тестовой выборке</b>
ResNet-34	15	0.1638
MobileNetV2	25	0.9249
EfficientNet-B0	62	0.9722

Как можно заметить из таблицы, лучше всего себя показала модель с энкодером на основе EfficientNet-B0, что ожидаемо, т.к. эта архитектура является более точной при тестировании на датасете ImageNet [10]. Но при этом обучение заняло больше времени, чем у модели с MobileNetV2. Модель с ResNet показала худший результат, а меньшее количество эпох обучения объясняется ранней остановкой обучения в связи с отсутствием улучшения метрики на валидационной выборке. Очевидно, что модель довольно быстро переобучилась, что можно будет увидеть далее на примерах сегментации лёгких на тестовой выборке.

Также важным параметром является размер модели, который также влияет на время вычисления – чем больше модель, тем больше ресурсов требуется. В таблице 2 представлены эти параметры для использованных моделей.

Таблица 2 – Размеры моделей

Энкодер	Размер модели, Мб
ResNet-34	283
MobileNetV2	96.3
EfficientNet-B0	121

Как мы видим самая маленькая модель – MobileNetV2, что неудивительно, ведь эта модель разрабатывалась для работы на мобильных устройствах. Тем не менее, модель на основе EfficientNet-B0 больше только на 25%, но при этом имеет большую точность. Модель на основе ResNet-34 в три раза больше MobileNetV2. На основе этого можно сделать вывод, что не всегда самая глубокая свёрточная нейронная сеть является самой точность.

Примеры сегментации легких на снимках из тестовой выборки представлены на рисунках 14-22 (а – оригинальное изображение, б – истинная маска, в – предсказанная маска, г – разница между истинной и предсказанной маской).

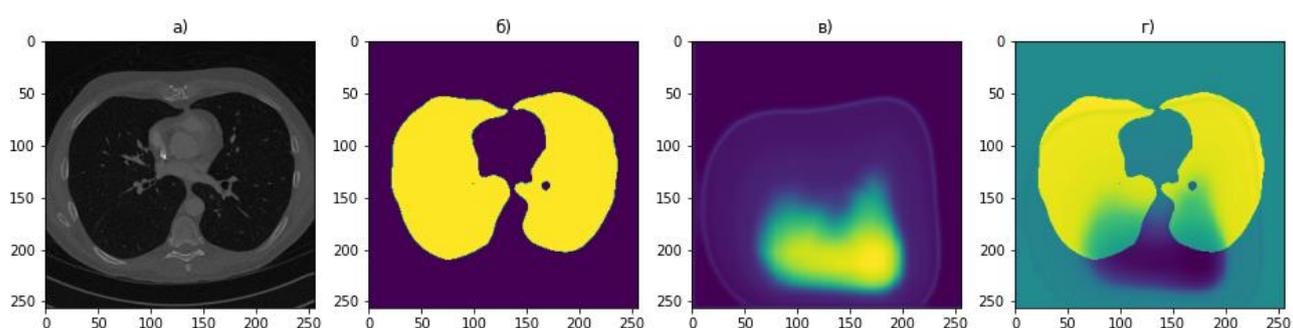


Рисунок 14 – Пример сегментации 1, ResNet-34

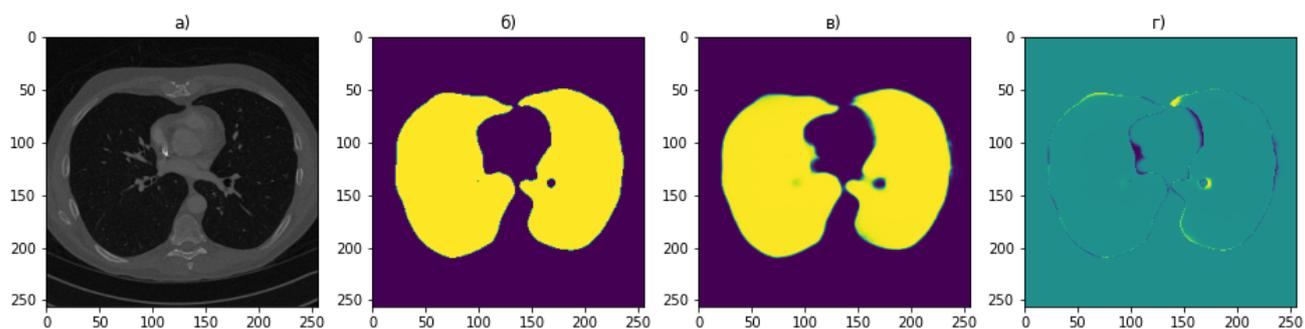


Рисунок 15 – Пример сегментации 1, MobileNetV2

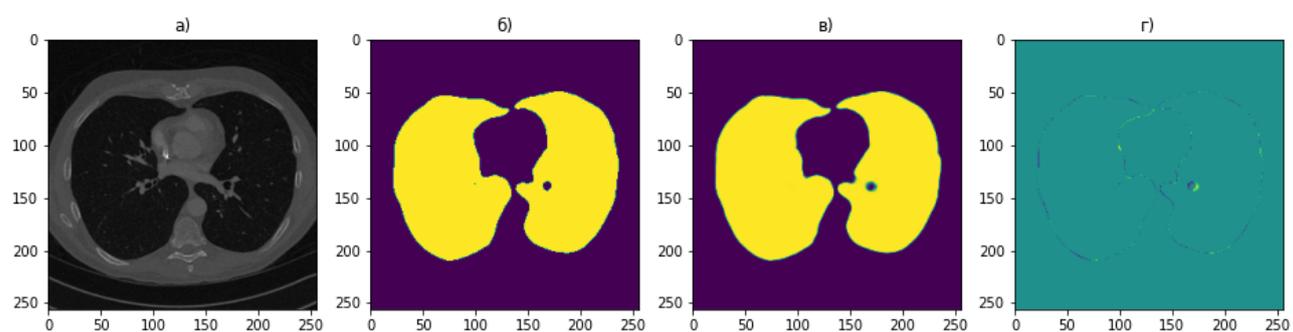


Рисунок 16 – Пример сегментации 1, EfficientNet-B0

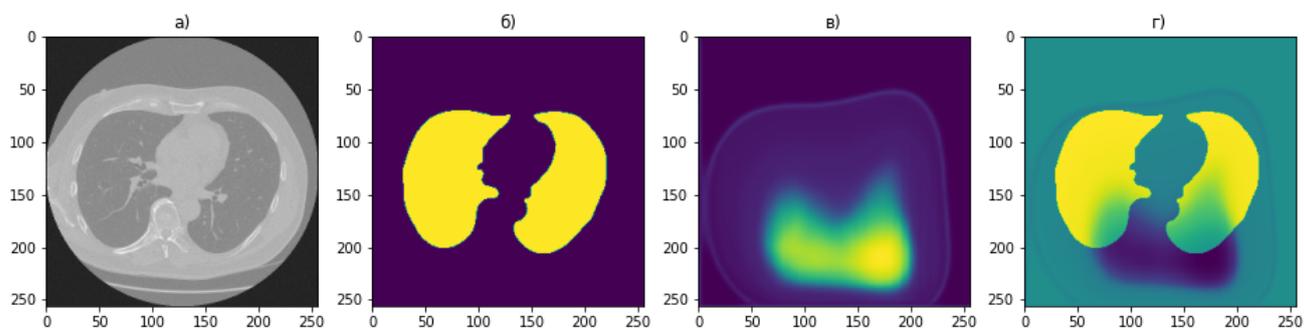


Рисунок 17 – Пример сегментации 2, ResNet-34

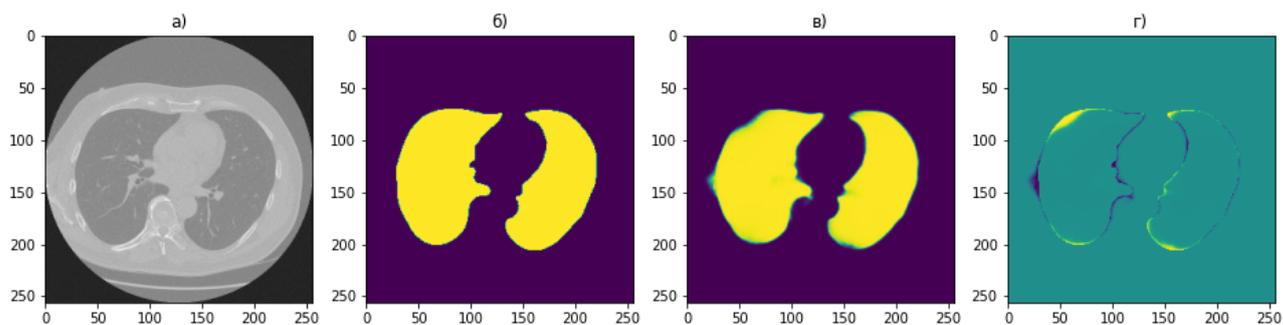


Рисунок 18 – Пример сегментации 2, MobileNetV2

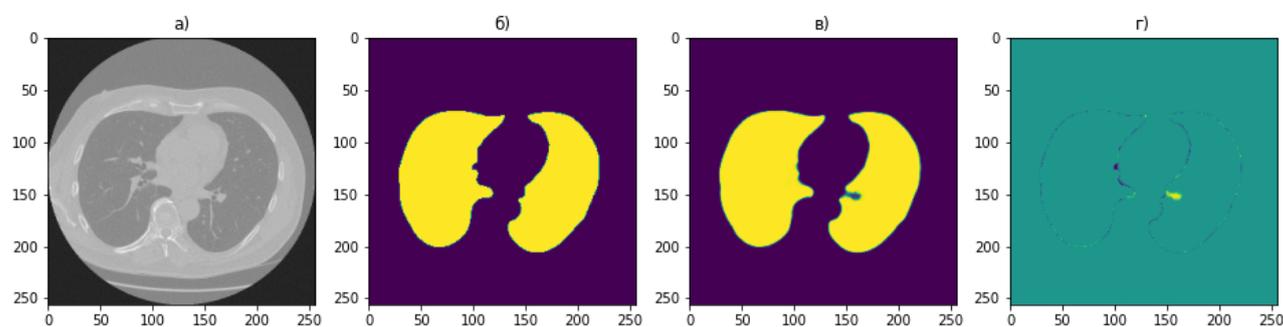


Рисунок 19 – Пример сегментации 2, EfficientNet-B0

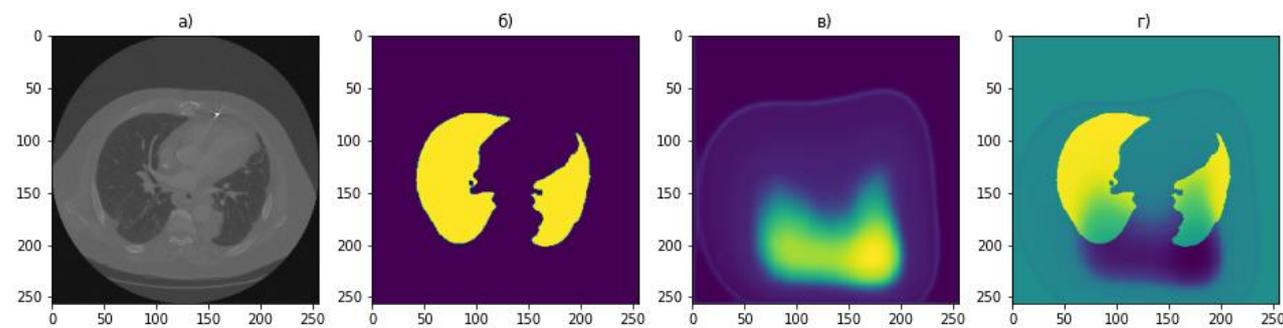


Рисунок 20 – Пример сегментации 3, ResNet-34

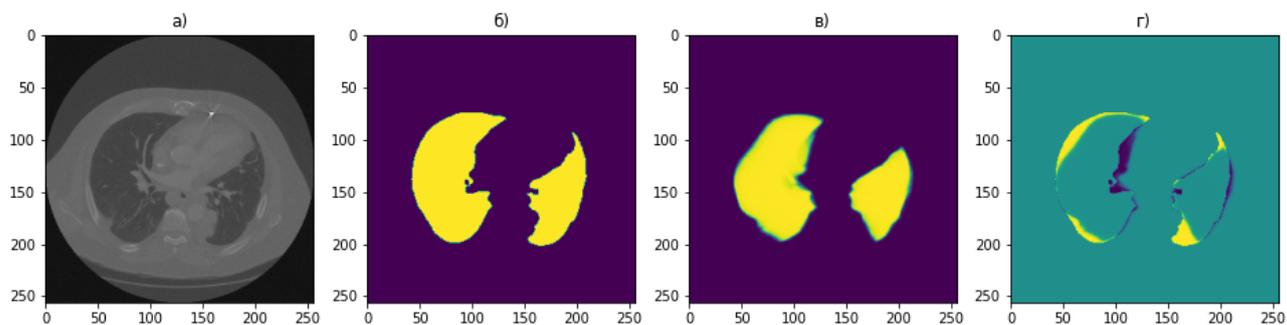


Рисунок 21 – Пример сегментации 3, MobileNetV2

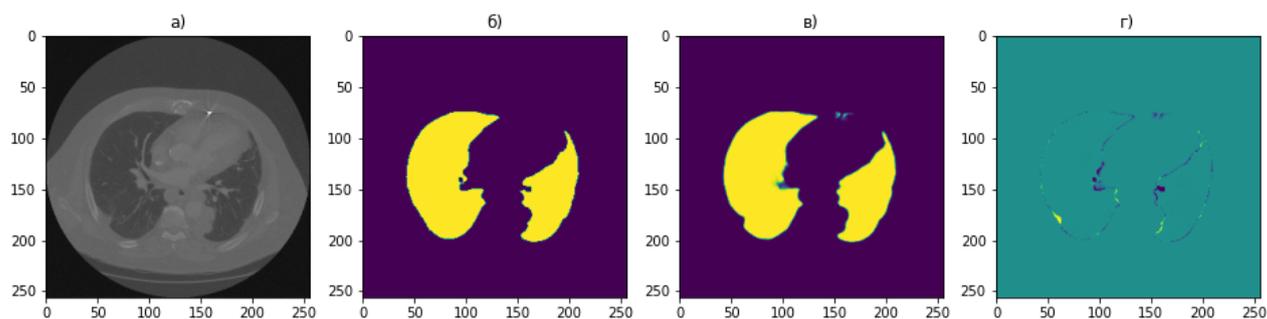


Рисунок 22 – Пример сегментации 3, EfficientNet-B0

Как можно заметить из примеров, модель на основе ResNet-34 действительно переобучилась, как было сказано ранее. Модель на основе MobileNetV2 делает более грубые предсказания, чем модель на основе EfficientNet-B0.

## **4 Финансовый менеджмент, ресурсоэффективность и энергосбережение**

### **4.1 Предпроектный анализ**

В рамках данной научно-исследовательской работы разрабатывается алгоритм сегментации лёгких на снимках компьютерной томографии. Данный алгоритм может использоваться в системах анализа снимков компьютерной томографии для облегчения и ускорения диагностики патологий врачами-рентгенологами. Также данный алгоритм может быть использован при разработке системы автоматизированного определения патологий. Потенциальными потребителями данной технологии являются организации здравоохранения. Также потенциальным потребителем данного исследования может являться научное сообщество, заинтересованное в использовании наработок по данной работе в рамках других исследований.

Для эффективного использования научного потенциала научно-исследовательского проекта необходимо не только провести исследовательскую работу, но и провести анализ исследования с точки зрения финансового менеджмента.

Задачами данного раздела являются:

1. Определение перспективности технологии с помощью технологии QuaD
2. Определение сильных и слабых сторон, а также возможностей и угроз с помощью SWOT-анализа
3. Оценка готовности проекта к коммерциализации
4. Определение целей и результата научно-исследовательского проекта
5. Планирование работы, распределение задач и определение трудоемкости
6. Формирование бюджета научно-исследовательского проекта

## 7. Проведение анализа рисков

### 4.2 Технология QuaD

В рамках данной работы анализ конкурентных технических решений невозможен ввиду отсутствия открытых данных о наличии и свойствах подобных разработок. Это связано с тем, что такие программные решения разрабатываются для нужд конкретной медицинской организаций либо являются частью более сложной медицинской системы.

Технология QuaD представляет собой гибкий инструмент измерения характеристик, описывающих качество новой разработки и её перспективность на рынке и позволяющие принимать решение целесообразности вложения денежных средств в научно-исследовательский проект. Составленная таблица QuaD представлена в таблице 3.

Таблица 3 – Оценочная карта технологии QuaD

Критерий оценки	Вес критерия	Баллы	Максимальный балл	Относительное значение	Средневзвешенное значение
Энергоэффективность	0,15	90	100	0,9	0,135
Помехоустойчивость	0,1	90	100	0,9	0,09
Надежность	0,15	95	100	0,95	0,1425
Унифицированность	0,05	80	100	0,8	0,04
Производительность алгоритма	0,05	65	100	0,65	0,0325
Пользовательский интерфейс	0,05	25	100	0,25	0,0125
Безопасность	0,1	100	100	1	0,1
Потребность в ресурсах памяти	0,1	75	100	0,75	0,075
Функциональная мощность	0,05	80	100	0,8	0,04
Простота эксплуатации	0,1	95	100	0,95	0,095
Качество интеллектуального интерфейса	0,05	80	100	0,8	0,04
Прозрачность кода	0,05	100	100	1	0,05
Итого	1				0,8525

Значение качества по технологии QuaD составило 0,8525. Можно сказать, что подобная разработка может считаться перспективной, так как у рассматриваемой разработки высокие показатели по всем наиболее важным критериям, таким как надежность, функциональная мощность и т.д.

### **4.3 SWOT-анализ**

SWOT – Strengths (сильные стороны), Weaknesses (слабые стороны), Opportunities (возможности) и Threats (угрозы) – это комплексный анализ научно-исследовательского проекта. Такой анализ применяют для исследования внешней и внутренней среды проекта. Составленная матрица SWOT представлена в таблице 4.

Таблица 4 – SWOT-анализ

<p><b>Сильные стороны научно-исследовательского проекта:</b></p> <p>С1. Наличие опыта работы с методами глубокого обучения.</p> <p>С2. Достаточный уровень развития технологий компьютерного зрения для решения задачи.</p> <p>С3. Большое количество библиотек с открытым исходным кодом.</p> <p>С4. Наличие промышленного опыта разработки ПО.</p>	<p><b>Слабые стороны научно-исследовательского проекта:</b></p> <p>Сл1. Высокая трудоемкость методов исследования.</p> <p>Сл2. Высокие требования к вычислительным ресурсам для обучения нейронной сети для алгоритма.</p> <p>Сл3. Дефицит размеченных данных для обучения нейронной сети.</p> <p>Сл4. Нехватка сотрудников для исследования.</p>
<p><b>Возможности:</b></p> <p>В1. Возможность применения наработок по исследованию в других областях.</p> <p>В2. Возможность использовать бесплатные вычислительные мощности, предоставляемые для студентов и исследователей.</p>	<p><b>Угрозы:</b></p> <p>У1. Невозможность внедрения из-за отрицательных результатов.</p> <p>У2. Несвоевременное финансирование исследования.</p> <p>У3. Увеличение сроков исследования из-за нехватки вычислительных ресурсов.</p>

Далее выявляются соответствия сильных и слабых сторон проекта условиям внешней среды. Это необходимо для оценки необходимости каких-либо изменений проекта. В таблице 5 представлены соответствия сильных сторон проекта и возможностей его реализации.

Таблица 5 – Сильные стороны исследования и возможности

Сильные стороны исследования					
Возможности исследования		C1	C2	C3	C4
	V1	+	+	+	-
	V2	-	-	-	+

Из таблицы 4 следует, что сильные стороны исследования соответствуют возможностям по следующим параметрам:

1. V1C1C3C3
2. V2C1

Возможность V1 соответствует сильным сторонами C1C3C4, т.к. возможность применения наработок в других областях зависит от наличия опыта работы с методами глубокого обучения, текущего уровня развития технологий, а также от наличия соответствующих библиотек с открытым исходным кодом.

Возможность V2 соответствует сильной стороной C4, т.к. наличие опыта разработки ПО позволяет с легкостью использовать бесплатные вычислительные ресурсы для академических целей.

В таблице 6 представлены соответствия слабых сторон проекта и возможностей его реализации.

Таблица 6 – Слабые стороны исследования и возможности

Слабые стороны исследования					
Возможности исследования		Сл1	Сл2	Сл3	Сл4
	В1	+	+	+	-
	В2	+	+	-	-

Из таблицы 6 следует, что сильные стороны исследования соответствуют возможностям по следующим параметрам:

1. В1Сл1Сл2Сл3
2. В2Сл2Сл4

Возможность В1 взаимодействует со слабыми сторонами Сл1Сл2Сл3, т.к. трудоемкость исследования, высокие требования к вычислительным ресурсам и нехватка размеченных данных могут помешать использовать наработки в других областях.

Возможность В2 взаимодействует со слабыми сторонами Сл1Сл2, т.к. трудоемкость исследования и высокие требования к вычислительным ресурсам не позволят полноценно использовать бесплатные вычислительные ресурсы.

В таблице 7 представлены соответствия слабых сторон проекта и возможностей его реализации.

Таблица 7 – Сильные стороны исследования и угрозы

Сильные стороны исследования					
Угрозы исследования		С1	С2	С3	С4
	У1	-	-	-	-
	У2	-	-	+	+
	У3	-	-	-	-

Из таблицы 7 следует, что сильные стороны исследования соответствуют угрозам по следующим параметрам:

1. У2С2С4

Угроза У2 взаимодействует с сильными сторонами С3С4, т.к. наличие опыта разработки ПО позволит не привлекать дополнительных специалистов, а наличие библиотек с открытым исходным кодом позволит сэкономить на покупке коммерческого ПО, что дает возможность начать исследование не дожидаясь финансирования.

В таблице 8 представлены соответствия слабых сторон проекта и возможностей его реализации.

Таблица 8 – Слабые стороны исследования и угрозы

Сильные стороны исследования					
		Сл1	Сл2	Сл3	Сл4
Угрозы исследования	У1	-	-	-	-
	У2	-	+	-	-
	У3	-	+	-	-

Из таблицы 8 следует, что сильные стороны исследования соответствуют угрозам по следующим параметрам:

1. У2Сл2
2. У3Сл2

Угрозы У2У3 взаимодействует со слабой стороной С2, т.к. высокие требования к вычислительным ресурсам могут сильно растянуть проект во времени с учетом угроз несвоевременного финансирования.

Анализ показал, что главными угрозами проекту являются запоздание исследования в связи с нехваткой вычислительных ресурсов и несвоевременное финансирование исследования. Слабыми сторонами проекта высокая трудоемкость методов исследования, высокие требования к вычислительным ресурсам для обучения нейронной сети для алгоритма, дефицит размеченных данных для обучения нейронной сети, нехватка сотрудников для исследования. Тем не менее, проект обладает и сильными сторонами, позволяющими

частично компенсировать слабые стороны, а также уменьшить негативное влияние возможных угроз.

#### **4.4 Оценка готовности научно-исследовательского проекта к коммерциализации**

Для научно-исследовательского проекта важно оценить степень его готовности к коммерциализации, а также оценить уровень собственных знаний для её проведения. По результатам такого анализа можно сделать вывод о готовности научно-исследовательского проекта к коммерциализации, а также о необходимости привлечения иных специалистов в команду проекта (таблица 9).

Таблица 9 – Бланк оценки степени готовности научно-исследовательского проекта к коммерциализации

<b>п/п</b>	<b>Наименование</b>	<b>Степень проработанности научно-исследовательского проекта</b>	<b>Уровень имеющихся знаний у разработчика</b>
1.	Определен имеющийся научно-технический задел	4	4
2.	Определены перспективные направления коммерциализации научно-технического задела	4	4
3.	Определены отрасли и технологии (товары, услуги) для предложения на рынке	4	4
4.	Определена товарная форма научно-технического задела для представления на рынок	4	4
5.	Определены авторы и осуществлена охрана их прав	2	3
6.	Проведена оценка стоимости интеллектуальной собственности	2	3
7.	Проведены маркетинговые исследования рынков сбыта	3	4
8.	Разработан бизнес-план коммерциализации научной	3	4

п/п	Наименование	Степень проработанности научно-исследовательского проекта	Уровень имеющихся знаний у разработчика
	разработки		
9.	Определены пути продвижения научной разработки на рынок	3	4
10.	Разработана стратегия (форма) реализации научной разработки	4	4
11.	Проработаны вопросы международного сотрудничества и выхода на зарубежный рынок	2	3
12.	Проработаны вопросы использования услуг инфраструктуры поддержки, получения льгот	2	3
13.	Проработаны вопросы финансирования коммерциализации научной разработки	3	4
14.	Имеется команда для коммерциализации научной разработки	2	4
15.	Проработан механизм реализации научно-исследовательского проекта	4	3
	<b>ИТОГО БАЛЛОВ</b>	46	55

Итоговый балл  $B_{\text{сум}}$  по степени проработанности научно-исследовательского проекта равен 46 баллам и по уровню имеющихся знаний у разработчика равен 55 баллам. Если значение  $B_{\text{сум}}$  лежит в интервале от 45 до 59 баллов, то перспективность такой разработки считается выше средней. Можно сделать вывод, что для дальнейшей коммерциализации научно-исследовательского проекта необходимо вовлечение сторонних специалистов в этой области и более глубокая проработка плана коммерциализации.

Успех продвижения товара на рынок во многом зависит от правильности выбора метода коммерциализации. Для данного научно-исследовательского

проекта наиболее приемлемым методом коммерциализации является создание собственного предприятия для реализации конечного продукта и его продажи медицинским организациям.

#### 4.5 Инициация научно-исследовательского проекта

Группа процессов инициации состоит из процессов, направленных на определение нового научно-исследовательского проекта. В рамках процессов инициации определяются цели, а также заинтересованные стороны проекта.

В данном разделе приводится информация о заинтересованных сторонах научно-исследовательского проекта, иерархии целей проекта и критериях достижения целей.

Таблица 10 содержит информацию о заинтересованных сторонах научно-исследовательского проекта.

Таблица 10 – Заинтересованные стороны научно-исследовательского проекта

<b>Заинтересованные стороны научно-исследовательского проекта</b>	<b>Ожидания заинтересованных сторон</b>
Отделение информационных технологий ТПУ	Научные публикации Защита магистерской диссертации
Медицинский персонал	Уменьшение времени работы с изображениями лёгких на снимках компьютерной томографии
Пациенты	Получение более качественных консультаций врачей, благодаря более точной диагностике
Научное сообщество	Алгоритм сегментации лёгких на снимках компьютерной томографии

В таблице 11 представлена информация об иерархии целей научно-исследовательского проекта и критериев их достижения.

Таблица 11 – Иерархия целей научно-исследовательского проекта и критерии их достижения

<b>Цели проекта:</b>	Разработать способ сегментации лёгких на снимках компьютерной томографии
<b>Ожидаемые результаты проекта:</b>	Алгоритм для сегментации лёгких на снимках компьютерной томографии
<b>Требования к результату проекта:</b>	Разработан способ обучения нейронной сети для точной сегментации лёгких
	Разработан алгоритм сегментации лёгких с небольшими требованиями к вычислительным ресурсам для работы на персональных компьютерах
	Бесперебойная работа программных модулей проекта
	Формализованное описание работы программных модулей проекта

## **4.6 Планирование научно-исследовательских работ**

### **4.6.1 Организационная структура научно-исследовательского проекта**

На данном этапе работы определяется состав рабочей группы научно-исследовательского проекта, определяются роли каждого участника в данном проекте. В таблице 12 определены участники научно-исследовательского проекта и их роли.

Таблица 12 – Рабочая группа научно-исследовательского проекта

№ п/п	ФИО	Роль в проекте	Функции
1	Аксёнов Сергей Владимирович	Научный руководитель	Составление научных задач, контроль выполнения проекта, проверка разработки, проверка документации
2	Войцеховский Алексей Алексеевич	Инженер	Проектирование, реализация

#### 4.6.2 Структура работ в рамках научного исследования

Для организации научно-исследовательской работы применяются различные методы экономического планирования с целью эффективного использования времени и рабочей силы, а также снижения трудозатрат. Планирование работы заключается в следующем:

1. Составление перечня работ, необходимых для достижения поставленной задачи;
2. Определение участников;
3. Установление продолжительности в рабочих днях;
4. Построение линейного графика и его оптимизации.

Примерный порядок этапов и работ, распределение исполнителей по данным видам работ приведен в таблице 13. Далее используются следующие сокращения:

1. НР – научный руководитель;
2. И – инженер.

Таблица 13 – Перечень этапов, работ и распределение исполнителей

Код работы	Название	Длительность, дни	Дата начала работ	Дата окончания работ	Состав участников (ФИО ответственных исполнителей)
1	Выбор научного руководителя	7	15.02.20	21.02.20	Войцеховский А.А.
2	Составление и утверждение темы	4	22.02.20	25.02.20	Аксёнов С.В.
3	Составление календарного плана-графика выполнения работы	3	26.02.20	28.02.20	Аксёнов С.В., Войцеховский А.А.
4	Подбор и изучение литературы по теме магистерской работы	14	01.03.20	14.03.20	Войцеховский А.А.
5	Анализ предметной области	7	15.03.20	21.03.20	Войцеховский А.А.
6	Подготовка данных	17	22.03.20	07.04.20	Войцеховский А.А.
7	Выбор архитектуры нейронной сети	10	08.04.20	17.04.20	Войцеховский А.А.
8	Обучение нейронной сети на размеченных данных	18	18.04.20	05.05.20	Войцеховский А.А.
9	Согласование выполненной работы с научным руководителем	7	06.05.20	12.05.20	Аксёнов С.В., Войцеховский А.А.
10	Выполнение других частей работы (финансовый менеджмент, социальная ответственность)	14	13.05.20	26.05.20	Войцеховский А.А.
11	Подведение итогов, оформление работы	5	27.05.20	31.05.20	Аксёнов С.В., Войцеховский А.А.
<b>ИТОГО:</b>		<b>106</b>	<b>15.02.20</b>	<b>31.05.20</b>	

#### 4.6.3 Определение трудоемкости выполнения работ

Наиболее ответственной частью экономических расчетов по теме является расчет трудоемкости работ, так как трудовые затраты составляют основную часть стоимости НИР. Под трудоемкостью работ понимают максимально допустимые затраты труда в человеко-днях на выполнение НИР с

учетом организационно технических мероприятий, обеспечивающих наиболее рациональное использование выделенных ресурсов.

Так как отсутствует нормативная база по проводимым работам, а также достоверная информация о процессе выполнения подобных работ иными исполнителями, воспользуемся экспертным способом оценки продолжительности выполнения запланированных работ.

Определим ожидаемое время проведения работ, длительность этапов в рабочих и календарных днях, по формулам, воспользовавшись формулой:

$$t_{ож} = \frac{3 \cdot t_{\min} + 2 \cdot t_{\max}}{5}, \text{ где:}$$

- 1)  $t_{ож}$  – ожидаемое время выполнения  $i$ -го этапа работ;
- 2)  $t_{\min}$  – минимально возможная трудоемкость выполнения заданной  $i$ -ой работы (в предположении наиболее благоприятного стечения обстоятельств);
- 3)  $t_{\max}$  – максимально возможная трудоемкость выполнения заданной  $i$ -ой работы (в предположении наиболее неблагоприятного стечения обстоятельств).

Ожидаемое, минимальное и максимальное время исполнения в предложенной выше формуле, оцениваются в рабочих днях на человека. Произведем перевод этих величин в календарные дни, воспользовавшись следующей формулой:

$$T_{КД} = T_{РД} \cdot T_K, \text{ где:}$$

- 1)  $T_{КД}$  – продолжительность выполнения этапа в календарных днях;
- 2)  $T_K$  – коэффициент календарности, позволяющий перейти от длительности работ в рабочих днях к их аналогам в календарных днях. Рассчитывается по формуле:

$$T_K = \frac{T_{KL}}{T_{KL} - T_{ВД} - T_{ПД}}, \text{ где:}$$

- 1)  $T_{KL}$  – календарные дни ( $T_{KL} = 365$ );
- 2)  $T_{ВД}$  – выходные дни ( $T_{ВД} = 52$ );
- 3)  $T_{ПД}$  – праздничные дни ( $T_{ПД} = 10$ ).

В свою очередь рабочие дни рассчитываются по следующей формуле:

$$T_{РД} = \frac{t_{ож}}{K_{ВН}} \cdot K_{Д}, \text{ где:}$$

- 1)  $K_{ВН}$  – коэффициент выполнения работ, учитывающий влияние внешних факторов на соблюдение предварительно определенных длительностей;

$K_{Д}$  – коэффициент, учитывающий дополнительное время на компенсацию непредвиденных задержек и согласование работ ( $K_{Д} = 1 - 1.2$ ; в этих границах конкретное значение принимает сам исполнитель).

Для простоты расчетов примем  $K_{Д}$  и  $K_{ВН}$ , равными единице. Тогда формула для расчета календарных дней преобразуется в следующую формулу:

$$T_{КД} = T_{РД} \cdot T_K = t_{ож} \cdot T_K = \frac{3 \cdot t_{\min} + 2 \cdot t_{\max}}{5}$$

Воспользовавшись данными из таблицы 11, приведенными выше формулами, произведем расчет продолжительности выполнения работ студентом в календарных днях. Результаты расчетов представлены в таблице 14.

Таблица 14 – Временные показатели проведения научного исследования

Название работы	Трудоемкость работ, чел-дни						Длительность работ в рабочих днях $T_{РД}$	Длительность работ в календарных днях $T_{КД}$		
	$t_{\min}$		$t_{\max}$		$t_{ож}$					
	Инженер	Научный руководитель	Инженер	Научный руководитель	Инженер	Научный руководитель	Одновременное выполнение работ		Одновременное выполнение работ	
							Инженер	Научный руководитель	Инженер	Научный руководитель
Выбор научного руководителя	5	-	8	-	6,2	-	6,2	-	7	-
Составление и утверждение темы	-	3	-	4	-	3,4	-	3,4	-	4
Составление календарного плана-графика выполнения работы	2	2	4	4	2,8	2,8	2,8	2,8	3	3
Подбор и изучение литературы по теме магистерской работы	11	-	13	-	11,8	-	11,8	-	14	-
Анализ предметной области	5	-	8	-	6,2	-	6,2	-	7	-
Подготовка данных	13	-	15	-	13,8	-	13,8	-	17	-
Выбор архитектуры нейронной сети	7	-	11	-	8,6	-	8,6	-	10	-
Обучение нейронной сети на размеченных данных	13	-	17	-	14,6	-	14,6	-	18	-
Согласование	5	5	8	8	6,2	6,2	6,2	6,2	7	7

выполненной работы с научным руководителем										
Выполнение других частей работы (финансовый менеджмент, социальная ответственность )	11	-	13	-	11,8	-	11,8	-	14	-
Подведение итогов, оформление работы	3	-	6	-	4,2	-	4,2	-	5	-
Итого							86,2	12,4	104	15

#### 4.6.4 Разработка графика проведения научного исследования

По данным из таблицы 14 создадим диаграмму Ганта (таблица 15).

Таблица 15 – Календарный план-график проведения НИОКР

№	Этап	Исполнитель	$T_{кд}$	Продолжительность выполнения работ													
				Февраль		Март		Апрель		Май		Июнь					
1	Выбор научного руководителя	И	7		///												
2	Составление и утверждение темы	НР	4		///												
3	Составление календарного плана-графика выполнения работы	И	3		///												
		НР	3		///												
4	Подбор и изучение литературы по теме магистерской работы	И	14				///										
5	Анализ предметной области	И	7				///										
6	Подготовка данных	И	17					///									
7	Выбор архитектуры нейронной сети	И	10						///								
8	Обучение нейронной сети на размеченных данных	И	18							///							
9	Согласование выполненной работы научным руководителем	И	7									///					
		НР	7									///					
10	Выполнение других частей работы (финансовый менеджмент, социальная	И	14										///				

	ответственность)																		
11	Подведение итогов, оформление работы	И	5																
		НР	5																

#### 4.7 Бюджет научно-технического исследования

Бюджет научно-технического исследования должен быть основан на достоверном отображении всех видов расходов, связанных выполнением проекта. В процессе формирования бюджета разработки используется следующая группировка затрат по статьям:

1. Материальные затраты разработки;
2. Основная заработная плата исполнителей темы;
3. Дополнительная заработная плата исполнителей темы;
4. Отчисления во внебюджетные фонды (страховые отчисления);
5. Затраты на научные и производственные командировки;
6. Накладные расходы.

##### 4.7.1 Расчет материальных затрат

Для вычисления материальных затрат воспользуемся следующей формулой:

$$Z_M = (1 + k_t) \cdot \sum_{i=1}^m C_i \cdot N_{расх.}, \text{ где:}$$

- 1)  $m$  – количество видов материальных ресурсов;
- 2)  $C_i$  – цена приобретения единицы  $i$ -го вида потребляемых материальных ресурсов;
- 3)  $N_{расх.}$  – количество материальных ресурсов  $i$ -го вида, планируемых к использованию при выполнении научного исследования;

4)  $k_t$  – коэффициент, учитывающий транспортно-заготовительные расходы.

Таблица 16 – Материальные затраты

Наименование	Единица измерения	Количество	Цена за единицу, руб.	Затраты на материалы, руб.
Лист А4	уп.	1	250	250
Ручка	шт.	3	60	180
<b>Итого, руб.</b>				<b>430</b>

#### 4.7.2 Расчет затрат на специальное оборудование для научных работ

В данном разделе учитываются затраты, связанные с приобретением оборудования, необходимого для проведения исследовательской работы. Затраты на приобретение оборудования для научной работы представлены в таблице 17.

Таблица 17 – Затраты на приобретение спецоборудования для научных работ

Наименование	Срок службы, год	Количество единиц оборудования	Цена, руб.
Персональный компьютер	3	1	49 000
<b>Итого, руб.</b>			<b>49 000</b>

#### 4.7.3 Расчет затрат на амортизацию оборудования

В данном разделе учитываются затраты, связанные с амортизационными отчислениями. Расчет отчислений ведется по следующей формуле:

$$I_{\text{ам.об.}} = \left( \frac{T_{\text{исп.об.}}}{365} \right) * K_{\text{об}} * N_{\text{ам}},$$

где  $T_{\text{ам.об.}}$  – время использования оборудования в календарных днях;

365 – число дней в году;

$K_{\text{об}}$  – стоимость оборудования;

$N_{ам}$  – норма амортизации.

Затраты на амортизацию оборудования для научной работы представлены в таблице 18. Срок использования в НТИ примем равным количеству рабочих дней работы инженера – 86,2 дня, т.к. вся работа подразумевает использование компьютера.

Таблица 18 – Затраты на амортизацию оборудования для научных работ

Наименование	Срок службы, год	Срок использования в НТИ, дн.	Цена, руб.	Амортизационные отчисления, руб.
Персональный компьютер	3	86,2	49 000	3 818,78
<b>Итого амортизационные отчисления, руб.</b>				<b>3 818,78</b>

#### 4.7.4 Основная заработная плата исполнителям работы

Статья включает основную заработную плату работников, непосредственно занятых выполнением НТИ, (включая премии, доплаты) и дополнительную заработную плату. Она рассчитывается по формуле

$$Z_{ЗП} = Z_{осн.} + Z_{доп.}, \text{ где:}$$

- 1)  $Z_{осн.}$  – основная заработная плата;
- 2)  $Z_{доп.}$  – дополнительная заработная плата (12-20 % от  $Z_{осн.}$ ).

Среднедневная заработная плата рассчитывается по формуле:

$$Z_{дн.} = \frac{Z_M \cdot M}{F_D}, \text{ где:}$$

- 1)  $Z_M$  – месячный должностной оклад работника, руб.;
- 2)  $M$  – количество месяцев работы без отпуска в течение года;
- 3)  $F_D$  – действительный годовой фонд рабочего времени научно-технического персонала, раб. дн.

Таблица 19 – Баланс рабочего времени

Показатели рабочего времени	Руководитель	Инженер
Календарное число дней	365	365
Количество нерабочих дней	52	120
- выходные дни	14	
- праздничные дни		
Потери рабочего времени		
- отпуск	48	24
- невыходы по болезни		
Действительный годовой фонд рабочего времени	251	221

Месячный оклад работника рассчитывается по формуле:

$$Z_M = Z_{TC} \cdot (1 + k_{np} + k_\delta) \cdot k_p, \text{ где:}$$

1)  $Z_{TC}$  – заработная плата по тарифной ставке, руб.;

2)  $k_{np}$  – премиальный коэффициент, равный 0,3 (т.е. 30% от  $Z_{TC}$ );

3)  $k_\delta$  – коэффициент доплат и надбавок составляет примерно 0,2 – 0,5 (в НИИ и на промышленных предприятиях – за расширение сфер обслуживания, за профессиональное мастерство, за вредные условия: 15-20% от  $Z_{TC}$ );

4)  $k_p$  – районный коэффициент, равный 1,3 (для Томска).

Расчёт основной заработной платы приведён в таблице 20.

Таблица 20 – Расчет основной заработной платы

Исполнители	Разряд	$k_{np}$	$k_{\delta}$	$k_p$	$Z_M$ , руб	$Z_{\text{осн.}}$ , руб.	$T_P$ , раб. дн.	$Z_{\text{осн.}}$ , руб.
Руководитель	1	0,3	0,2	1,3	65 644,8	2 719,94	12,4	33 727,30
Инженер		0	0	1,3	28 288	1 433,60	86,2	123 576,32
<b>Итого <math>Z_{\text{осн.}}</math></b>								<b>157 303,62</b>

#### 4.7.5 Дополнительная заработная плата

Дополнительная заработная плата включает заработную плату за не отработанное рабочее время, но гарантированную действующим законодательством. Расчет дополнительной заработной платы ведется по формуле:

$$Z_{\text{доп.}} = k_{\text{доп.}} \cdot Z_{\text{осн.}}, \text{ где:}$$

1)  $k_{\text{доп.}}$  – коэффициент дополнительной заработной платы (на стадии проектирования принимается равным 0,12 – 0,15).

$k_{\text{доп.}}$  равен 0,12. Результаты по расчетам дополнительной заработной платы сведены в таблицу 21.

Таблица 21 – Затраты на дополнительную заработную плату

Исполнители	Основная зарплата (руб.)	$k_{\text{доп.}}$	Дополнительная зарплата (руб.)
Руководитель	31 132,9	0,12	4 047,28
Инженер	123 576,32	0,12	14 829,16
<b>Итого</b>			<b>18 876,43</b>

#### 4.7.6 Отчисления во внебюджетные фонды

Величина отчислений во внебюджетные фонды определяется исходя из формулы:

$$Z_{\text{внеб.}} = k_{\text{внеб.}} \cdot (Z_{\text{осн.}} + Z_{\text{доп.}})$$

где  $k_{\text{внеб}}$  – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и пр.).

На 2020 г. в соответствии с Федеральным законом от 24.07.2009 №212-ФЗ установлен размер страховых взносов равный 28%. На основании пункта 1 ст.58 закона №212-ФЗ для учреждений, осуществляющих образовательную и научную деятельность в 2020 году пониженная ставка – 28%. Отчисления во внебюджетные фонды представлены в таблице 22.

Таблица 22 – Отчисления во внебюджетные фонды

Исполнитель	Основная заработная плата, руб.	Дополнительная заработная плата, руб.
Руководитель проекта	33 727,30	4 047,28
Инженер	123 576,32	14 829,16
Коэффициент отчислений во внебюджетные фонды	28%	
<b>Итого</b>		
Руководитель	10 576,88	
Инженер	38 753,53	
<b>Итого</b>	<b>49 330,42</b>	

#### 4.7.7 Накладные расходы

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов: печать и ксерокопирование материалов, оплата услуг связи, электроэнергии и т.д. Расчет накладных расходов определяется по формуле:

$$Z_{\text{нак.}} = \sum Cm \cdot k_{\text{нр}}, \text{ где:}$$

- 1)  $k_{\text{нр}}$  – коэффициент, учитывающий накладные расходы;
- 2)  $Cm$  – затраты по статьям накладных расходов.

Величину коэффициента накладных расходов можно взять в размере 16%.

$$Z_{\text{нак.}} = (430 + 49\,000 + 3\,818,78 + 157\,303,62 + 18\,876,43 + 49\,330,42) \\ * 0,16 = 278\,759,26 * 0,16 = 44\,601,48 \text{руб.}$$

#### 4.7.8 Контрагентные расходы

Контрагентные расходы включают затраты, связанные с выполнением каких-либо работ по теме сторонними организациями (контрагентами, субподрядчиками). В данном проекте отсутствует необходимость в стороннем подрядчике.

#### 4.7.9 Формирование бюджета затрат научно-исследовательского проекта

Рассчитанная величина затрат научно-исследовательской работы является основой для формирования бюджета затрат проекта. Определение бюджета затрат на научно-исследовательский проект по каждому варианту исполнения приведен в таблице 23.

Таблица 23 – Расчет бюджета затрат НИИ

Наименование статьи	Сумма, руб.
1. Материальные затраты НИИ	430
2. Затраты на приобретение спецоборудования для научных работ	49 000
3. Затраты на амортизационные отчисления	3 818,78
4. Затраты на заработную плату научному руководителю	37 774,58
5. Затраты на заработную плату инженеру	138 405,48
6. Затраты на отчисления во внебюджетный фонд	49 330,42
6. Накладные расходы	44 601,48
<b>Бюджет затрат НИИ</b>	<b>323 360,74</b>

## 4.8 Риски научно-исследовательского проекта

При разработке научно-исследовательского проекта следует понимать и учитывать возможные риски. Таблица 24 содержит результаты анализа возможных рисков.

Таблица 24 – Реестр рисков

№	Риск	Потенциальное воздействие	Вероятность наступления (1-5)	Влияние риска (1-5)	Уровень риска	Способы смягчения риска	Условия наступления
1	Кадровый риск	Отсутствие заинтересованных исполнителей проекта	3	4	Высокий	Повышение мотивации исполнителей проекта	Потеря интереса исполнителей к деятельности проекта
2	Технический риск	Потеря файлов проекта	1	5	Высокий	Регулярное создание резервных копий файлов проекта	Отказ используемого оборудования
3	Доступ к данным	Отсутствие данных для работы системы	1	4	Средний	Заключение официального договора на доступ к данным	Отсутствие размеченных данных
4	Недостаточная производительность алгоритма	Неконкурентоспособность разработки	1	3	Средний	Проведение сравнительного тестирования, прототипирование	Ошибки при постановке задачи, недостаточный анализ качества разработки и ее перспективности на рынке

Из анализа реестра рисков можно заметить, что первый и второй вид рисков встречается во многих проектах. Риск же отсутствия доступа к размеченным данным во время выполнения данного научно-исследовательского проекта существенен для реализации, однако маловероятен. Также существует риск, что разработанный алгоритм будет недостаточно эффективен, но он также маловероятен при должном тестировании в процессе разработки прототипа.

#### **4.9 Выводы по разделу финансовый менеджмент**

Данный исследовательский проект является перспективным согласно расчету с использованием технологии QuaD. Согласно SWOT-анализу проект имеет достаточно сильных сторон для использования возможностей и противодействия угрозам проекта. Для коммерциализации научно-исследовательского проекта необходимо вовлечение сторонних специалистов в этой области и создание собственного предприятия для разработки продукта. Основные работы в рамках данного научно-исследовательского проекта проводились в период с 15 февраля по 31 мая 2020 года. Команда проекта состоит из инженера и научного руководителя.

Бюджет проекта составляет 323 360,74 рублей. Большая часть расходов представляет собой затраты на ФОТ, что вполне соответствует структуре затрат проектов в сфере информационных технологий.

Все рассмотренные риски научно-исследовательского проекта являются существенными для его реализации, однако вероятность их наступления достаточно мала.

Использование алгоритма сегментации лёгких позволит автоматизировать диагностику заболеваний лёгких, тем самым увеличив её скорость и точность.

## **5 Социальная ответственность**

### **5.1 Введение**

Объектом исследования является алгоритм сегментации лёгких на снимках компьютерной томографии.

Научно-исследовательская работа заключалась в разработке алгоритма сегментации лёгких на снимках томографии. Исследование проводилось при помощи ЭВМ.

Данное исследование проводилось на базе ОИТ ИШИТР ТПУ, в аудитории 415, корпус КЦ ТПУ.

В данном разделе рассматриваются вопросы, связанные с обеспечением безопасности при разработке алгоритма. В связи с этим рассмотрены следующие вопросы:

1. Выявление вредных и опасных факторов;
2. Обеспечение экологической безопасности;
3. Обеспечение безопасности в чрезвычайных ситуациях;
4. Правовые и организационные вопросы обеспечения безопасности.

### **5.2 Правовые и организационные вопросы обеспечения безопасности**

Согласно Статьям 91 и 108 ТК РФ, регламентированное время продолжительности рабочего времени не должно превышать 40 часов в неделю, в течение рабочего дня работодатель обязан предоставить работнику перерыв для отдыха и питания. Предоставляемое работнику время для отдыха и питания располагается во временном промежутке от 30 минут до 2 часов, в зависимости от работодателя [21].

Рабочее место должно быть организовано с учетом эргономических требований согласно ГОСТ 12.2.032-78 «Система стандартов безопасности

труда. Рабочее место при выполнении работ сидя [22]. Общие эргономические требования» и ГОСТ 12.2.061-81 «ССБТ. Оборудование производственное. Общие требования безопасности к рабочим местам» [23].

Рабочий стол может быть любой конструкции, отвечающей современным требованиям эргономики и позволяющей удобно разместить на рабочей поверхности оборудование с учетом его количества, размеров и характера выполняемой работы. При отсутствии регулировки высота стола должна быть в пределах от 680 до 800 мм.

Рабочие места с персональными компьютерами по отношению к световым проемам должны располагаться так, чтобы естественный свет падал сбоку, желательно слева.

Схемы размещения рабочих мест с персональными компьютерами должны учитывать расстояния между рабочими столами с мониторами: расстояние между боковыми поверхностями мониторов не менее 1,2 м, а расстояние между экраном монитора и тыльной частью другого монитора не менее 2,0 м [24].

Быстрое и точное считывание информации обеспечивается при расположении плоскости экрана ниже уровня глаз пользователя, предпочтительно перпендикулярно к нормальной линии взгляда (нормальная линия взгляда 15 градусов вниз от горизонтали).

Поверхность пола в помещениях эксплуатации компьютеров должна быть ровной, без выбоин, нескользкой, удобной для очистки и влажной уборки, обладать антистатическими свойствами.

Кроме того, в помещении должны находиться аптечка первой медицинской помощи, углекислотный огнетушитель для тушения пожара.

Обязанности по обеспечению безопасных условий и охраны труда возлагаются на работодателя и отражены в статье 212 ТК РФ [21].

### 5.3 Профессиональная социальная безопасность

В аудитории, где находятся различные электроприборы и ЭВМ, могут возникнуть следующие вредные факторы: наличие некомфортных метеоусловий, производственного шума, недостаточной освещенности [25]. Также существует вероятность поражения электрическим током.

Таблица 25 – Возможные опасные и вредные факторы

Факторы (ГОСТ 12.0.003-2015) [5]	Этап работы		Нормативные документы
	Разработка	Эксплуатация	
1. Отклонение показателей микроклимата	+	+	СанПиН 2.2.4.548-96 Гигиенические требования к микроклимату производственных помещений [26].
2. Превышение уровня шума	+	+	СН 2.2.4/2.1.8.562–96. Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории застройки [27].
3. Недостаточная освещенность рабочей зоны	+	+	СП 52.13330.2016 Естественное и искусственное освещение. Актуализированная редакция СНиП 23-05-95* [28].
4. Психофизиологические факторы при работе с компьютером	+	+	СанПиН 2.2.2/2.4.1340-03. Гигиенические требования к персональным электронно-вычислительным машинам и организации работы [24].
5. Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека	+	+	ГОСТ 12.1.019-2017 ССБТ. Электробезопасность. Общие требования и номенклатура видов защиты [29].

### 5.3.1 Отклонение показателей микроклимата

Микроклимат помещения, как правило, характеризуют тремя основными параметрами:

- температура окружающего воздуха, °С;
- относительная влажность воздуха, %;
- скорость движения воздуха, м/с;

Неблагоприятные метеоусловия в помещении могут вызывать перенапряжение механизмов терморегуляции, перегрев или переохлаждение организма, что в свою очередь негативно влияет здоровье и продуктивность труда сотрудника.

В таблице 26 представлены оптимальные параметры микроклимата согласно СанПиН 2.2.4.548–96 в помещениях для легкой категории работ, к которой относятся работы, выполняемые сидя и не требующие физического напряжения. Работа персонала в данном случае относится к категории работ Ia, с интенсивностью энергозатрат до 139 ккал/ч. Работы производятся сидя и сопровождаются незначительным физическим напряжением [26].

Таблица 26 – Оптимальные параметры микроклимата

Период года	Температура, °С	Относительная влажность, %	Скорость движения воздуха, м/с, не более
Холодный	22-24	40-60	0,1
Теплый	23-25	40-60	0,1

К допустимым микроклиматическим нормам относят величины, которые способны вызвать изменение теплового состояния организма. При этом не возникает нарушений состояния здоровья, но могут ощущаться дискомфорт, ухудшение самочувствия и понижение работоспособности. Допустимые нормы устанавливаются в тех производственных помещениях, где по каким-либо причинам нет возможности обеспечить оптимальные условия.

Допустимые параметры для легкой категории работ представлены в таблице 27 [26].

Таблица 27 – Допустимые параметры микроклимата

Период года	Температура, °С		Относительная влажность, %	Скорость движения воздуха, м/с, не более	
	Ниже оптимальных значений	Выше оптимальных значений		Ниже оптимальных значений	Выше оптимальных значений
Холодный	20-21	24,1-25	15-75	0,1	0,1
Теплый	21-22	25,1-26	15-75	0,1	0,1

Для обеспечения оптимальных и допустимых показателей микроклимата в холодный период года следует применять средства защиты рабочих мест от остекленных поверхностей оконных проемов, чтобы не было охлаждения, а также необходимо установить отопительную систему. В теплый период года необходимо предусмотреть защиту от попадания прямых солнечных лучей. К средствам коллективной защиты можно отнести установки кондиционирования воздуха, основная задача которых поддерживать параметры воздуха в установленных пределах, для обеспечения надежной работы и комфортных условий для работников.

### 5.3.2 Превышение уровня шума

Шум – беспорядочное сочетание различных по частоте и силе звуков, мешающих человеческой деятельности и вызывающих неприятные ощущения.

Источниками шума являются колеблющиеся тела, звуковые волны от которых вызывают периодическое повышение или понижение давления воздуха. Ухо человека способно реагировать на изменения от  $2 \cdot 10^{-5}$  до 200 Па – эти величины являются пороговыми. Верхний порог слышимости также называют болевым порогом.

Продолжительное воздействие шума может привести к шумовой болезни – заболеванию организма, которое сопровождается поражением органов слуха и центральной нервной системы. Также шум вызывает головную боль, раздражительность, быструю утомляемость, потерю слуха, повышение артериального давления.

Допустимые шумовые характеристики рабочих мест регламентируются СанПиН 2.2.4/2.1.8.562-96 и представлены в таблице 28 [27].

Таблица 28 – Допустимые уровни звукового давления

Полоса частот, Гц	1,5	63	125	250	500	1000	2000	4000	8000
Уровни звукового давления, дБ	86	71	61	54	49	45	42	40	38
Уровни звука, дБА	50								
Эквивалентные уровни звука, дБА	50								

Работа в данном случае относится к напряженной 2 степени (творческая работа, решение сложных задач) и максимальный уровень шума не должен превышать 50 дБА [27]. Это условие соблюдается на рабочем месте за компьютером. Таким образом, рабочее место удовлетворяет требованиям по максимальному шуму.

### 5.3.3 Освещение

Освещение на рабочем месте должно отвечать условиям работы и быть оптимальным по величине. Спектр освещения должен стремиться к спектру дневного света, т.к. физиология человека лучше адаптирована к последнему. В случае недостатка освещения человек быстро утомляется, снижается работоспособность, возрастает опасность ошибок и несчастных случаев. Слишком высокая освещенность, как и недостаточная, вызывает быстрое

утомление глаз и снижение остроты зрения. Также освещенность должна быть равномерной по площади [30].

В итоге требования к производственному освещению сводятся к следующим:

1. Величина освещенности должна соответствовать гигиеническим нормам, учитывающим условия зрительной работы;
2. Состав спектра искусственного света, должен стремиться к спектру естественного освещения;
3. Должна быть обеспечена равномерность уровня освещенности в помещении, чтобы избежать переутомления зрения.

Произведем расчёт освещения для аудитории 415 в Кибернетическом Центре ТПУ, в которой выполняется работа над научно-исследовательской работой. Расчёт общего равномерного искусственного освещения горизонтальной рабочей поверхности выполняется методом коэффициента светового потока, учитывающим отраженный от потолка и стен световой поток. Длина помещения  $A = 6$  м; ширина  $B = 5$  м; высота  $H = 3$  м; высота рабочей поверхности над полом  $h_p = 0,8$  м.

Рассчитаем площадь помещения:

$$S = A \cdot B = 6 \cdot 5 = 30 \text{ м}^2.$$

В помещении шесть потолочных светильников с четырьмя люминесцентными лампами с мощностью по 18 Вт (18x4) и световым потоком 1200 Лм. Длина светильника – 595 мм, ширина – 595 мм.

Критерий оптимальности расположения светильников – величина  $\lambda$ , для люминесцентных светильников с защитной решеткой лежит в диапазоне 1,1...1,3. Принимаем  $\lambda = 1,1$ . Расстояние светильников от перекрытия  $h_c = 0,2$  м.

Высота светильника над рабочей поверхностью определяется по формуле:

$$h = H - h_p - h_c = 3 - 0,8 - 0,2 = 2 \text{ м.}$$

Расстояние между соседними светильниками определяется по формуле:

$$L = \lambda \cdot h = 1,1 \cdot 2 = 2,2 \text{ м.}$$

Число рядов светильников в помещении:

$$N_b = B / L = 5 / 2,2 \approx 2.$$

Число светильников в ряду:

$$N_a = A / L = 6 / 2,53 \approx 3.$$

Общее число светильников:

$$N = N_a \cdot N_b = 3 \cdot 2 = 6.$$

Расстояние от рядов светильников до стены определяется по формуле:

$$l = L / 3 = 2,2 / 3 = 0,7 \text{ м.}$$

На рисунке 23 изображен план помещения и размещения светильников с люминесцентными лампами.

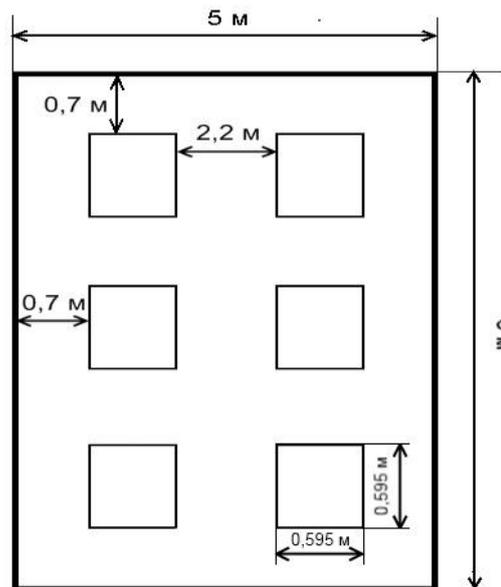


Рисунок 23 – План размещения светильников в аудитории

Индекс помещения определяется по формуле:

$$i = \frac{A \cdot B}{h \cdot (A + B)} = \frac{6 \cdot 5}{2 \cdot (6 + 5)} = 1,4$$

Определим коэффициент использования светового потока для рассматриваемой аудитории. Потолок белый (гипс), следовательно коэффициент отражения примем  $\rho_{\text{П}} = 70\%$  [38]. Стены средне-светлые (светло-серая краска), следовательно примем коэффициент отражения стен  $\rho_{\text{С}} = 50\%$  [38]. Пол в аудитории темный (темная краска), значит коэффициент отражения

пола примем равным  $\rho_p = 30\%$  [38]. Тогда используя таблицу коэффициентов использования для потолочных светильников [39] при коэффициентах  $\rho_{\Pi} = 70\%$ ,  $\rho_c = 50\%$ ,  $\rho_p = 30\%$  и индексе помещения  $i = 1,4$  коэффициент использования светового потока равен  $\eta \approx 0,52$ .

Требуемый световой поток люминесцентной лампы светильника определяется по формуле:

$$\Phi_{\Pi} = \frac{E \cdot A \cdot B \cdot K_3 \cdot Z}{n \cdot N \cdot \eta},$$

где  $E$  – нормируемое значение освещенности;

$A$  – длина помещения;

$B$  – ширина помещения;

$K_3$  – коэффициент запаса, учитывающий загрязнений и старение ламп, примем его равным  $K_3 = 1,5$ ;

$Z$  – коэффициент неравномерности освещения, для люминесцентных ламп он принимается равным  $Z = 1,1$  [39];

$n$  – число ламп в светильнике;

$N$  – число светильников;

$\eta$  – коэффициент использования светового потока.

Нормируемое значение освещенности для кабинетов и офисов при общем освещении принимается равным 300 лк [30]. Тогда, подставив все значения в формулу, получаем значение требуемой освещенности от одной лампы:

$$\Phi_{\Pi} = \frac{E \cdot A \cdot B \cdot K_3 \cdot Z}{n \cdot N \cdot \eta} = \frac{300 \cdot 6 \cdot 5 \cdot 1,5 \cdot 1,1}{4 \cdot 6 \cdot 0,52} = 1189,9 \text{ Лм.}$$

Далее необходимо проверить, что система освещения в аудитории соответствует требуемой освещенности. Для этого разница между световым потоком от одной лампы и требуемой величиной должна быть не менее -10% и не 20%. Воспользуемся формулой:

$$\frac{\Phi_{\text{ЛД}} - \Phi_{\Pi}}{\Phi_{\text{ЛД}}},$$

где  $\Phi_{\text{ЛД}}$  – световой поток от одной лампы;

$\Phi_{\text{П}}$  – требуемый световой поток от одной лампы.

Подставим значения и рассчитаем:

$$\frac{\Phi_{\text{ЛД}} - \Phi_{\text{П}}}{\Phi_{\text{ЛД}}} \cdot 100\% = \frac{1200 - 1198,9}{1200} \cdot 100\% = 0,092 \%$$

Таким образом,  $-10\% \leq 0,092\% \leq 20\%$  – необходимый световой поток лампы не выходит за пределы требуемого диапазона.

В помещении, в котором проводится исследование, соблюдены все требования к освещению, приведенные в СанПиН 2.2.1/2.1.1.1278-03, и имеются искусственные и естественные источники освещения [30].

### **5.3.4 Психофизиологические факторы при работе с компьютером**

При разработке алгоритма сотрудник сталкивается с напряженной зрительной работой и умственной нагрузкой, т.к. большая часть рабочего времени занята работой за персональным компьютером. В процессе работы с компьютером необходимо соблюдать правильный режим труда и отдыха. В противном случае у персонала отмечаются значительное напряжение зрительного аппарата с появлением жалоб на неудовлетворенность работой, головные боли, раздражительность, нарушение сна, усталость и болезненные ощущения в глазах.

Разработка алгоритма относится к творческой работе в режиме диалога с ПЭВМ (группа В) и нормируется по времени работы. Работа за компьютером занимает не более 4 часов времени, таким образом, исследование можно отнести ко II категории работы с ПЭВМ. Значит, регламентированное время перерывов составляет 70 минут. Во время перерывов рекомендуется выполнять гимнастику для глаз, позволяющую снять зрительное напряжения [24].

Также при работе за компьютером человек находится в фиксированной позе, которая не является физиологически естественной, что вредно для опорно-двигательного аппарата, а также приводит к застоям крови в сосудах и внутренних органах и другим заболеваниям. Для уменьшения негативного влияния монотонности труда рекомендуется во время перерывов заниматься какой-нибудь физической активностью: походить, размяться и т.д. [24].

Рабочее место, при соблюдении правил работы за ПК, полностью соответствует требованиям для нормального психического и физиологического функционирования человека.

### **5.3.5 Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека**

В процессе разработки алгоритма возможно поражение электрическим током, проходящим от сети к ПК и другого оборудования. В зависимости от условий в помещении опасность поражения человека электрическим током увеличивается или уменьшается. Не стоит работать с электрооборудованием в условиях повышенной влажности (длительное время более 75%), высокой температуры (более 35°C), наличии токопроводящей пыли, токопроводящих полов и возможности одновременного соприкосновения к имеющим соединение с землей металлическим элементам и металлическим корпусом электрооборудования. Исходя из этого, работа с электрооборудованием может проводиться только в помещениях без повышенной опасности, и возможность поражения током может быть только при прикосновении непосредственно с элементами ПК и токоведущим частям электроприборов [29].

Обязательны следующие меры предосторожности:

1. Перед началом работы нужно убедиться, что выключатели и розетка закреплены и не имеют оголенных токоведущих частей;

2. При обнаружении неисправности оборудования и приборов необходимо не делая никаких самостоятельных исправлений сообщить ответственному за оборудование;

3. Запрещается загромождать рабочее место лишними предметами.

К мероприятиям по предотвращению возможного поражения электрическим током [29]:

1. Соблюдение соответствующих расстояний до токоведущих частей или путем закрытия, ограждения токоведущих частей;

2. Применение блокировки аппаратов и ограждающих устройств для предотвращения ошибочных операций и доступа к токоведущим частям;

3. Применение предупреждающей сигнализации, надписей и плакатов;

4. Применение устройств для снижения напряженности электрических и магнитных полей до допустимых значений;

5. Использование средств защиты и приспособлений. Повышение электробезопасности в установках достигается применением систем защитного заземления, защитного зануления и защитного отключения [31].

Все требуемые меры защиты соблюдаются в помещении, где ведется исследование свойств материала.

## **5.4 Экологическая безопасность**

Охрана окружающей среды – комплексная проблема и наиболее активной формой ее решения является сокращение вредных выбросов промышленных предприятий через полный переход к безотходным или малоотходным технологиям производства.

Для уменьшения негативного воздействия на окружающую среду необходимо заранее рассмотреть варианты утилизации неисправного и устаревшего оборудования, перегоревших люминесцентных ламп. Утилизация

электрооборудования должна производиться в специализированных учреждениях [32].

### **5.5 Безопасность в чрезвычайных ситуациях**

Наиболее вероятной чрезвычайной ситуацией при исследовании материала является пожар на рабочем месте. Причиной пожара могут служить короткое замыкание, перегрузки по току, статическое электричество и т.д. В качестве противопожарных мероприятий должны быть применены следующие меры [33]:

1. Должно использоваться только исправное оборудование;
2. Должны проводиться периодические инструктажи по пожарной безопасности;
3. Пути и проходы для эвакуации людей должны содержаться в свободном состоянии;
4. В помещении должны находиться средства тушения пожара, средства связи;
5. Электрическая проводка электрооборудования и осветительных приборов должна быть исправна;
6. Все сотрудники должны знать место нахождения средств пожаротушения и уметь ими воспользоваться, средств связи и номера экстренных служб.

План эвакуации из аудитории, где проводится исследование, представлен на рисунке 24.

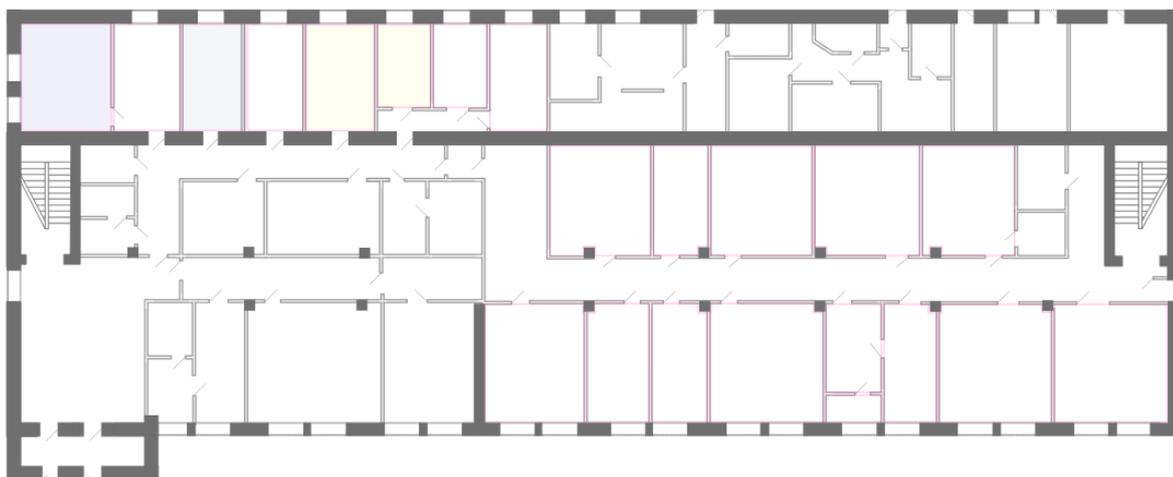


Рисунок 24 – План эвакуации

В связи с возможностью возникновения пожара разработан следующий план действий [33]:

1. В случае возникновения пожара сообщить о нем руководителю, постараться устранить очаг возгорания имеющимися силами при помощи первичных средств пожаротушения (огнетушитель порошковый, углекислотный О-1П0 (з)-АВСЕ) [14];

2. Привести в действие ручной пожарный извещатель, если очаг возгорания потушить не удастся;

3. Сообщить о возгорании в службу пожарной охраны по телефону 01 или 010, сообщить адрес, место и причину возникновения пожара;

4. Принять меры по эвакуации людей и материальных ценностей;

5. Встретить пожарную охрану, при необходимости сообщить всю необходимую информацию и оказать помощь при выборе наилучшего подхода к очагу возгорания.

По взрывопожарной и пожарной опасности помещения подразделяются на категории А, Б, В1 - В4, Г и Д, а здания – на категории А, Б, В, Г и Д [15].

Согласно НПБ 105-03 помещение, где производится разработка алгоритма, относится к категории В – горючие и трудногорючие жидкости, твердые горючие и трудногорючие вещества и материалы (в том числе пыли и волокна), вещества и материалы, способные при взаимодействии с водой,

кислородом воздуха или друг с другом только гореть, при условии, что помещения, в которых они имеются в наличии или обращаются, не относятся к категориям А или Б [36].

По степени огнестойкости данное здание относится к 1-й степени огнестойкости по СНиП 21-01-97 – здания с несущими и ограждающими конструкциями из естественных или искусственных каменных материалов, бетона или железобетона с применением листовых и плитных негорючих материалов (выполнено из кирпича, которое относится к трудно сгораемым материалам) [35].

В здании места работы соблюдены требования пожаробезопасности и имеются необходимые средства пожаротушения.

## **5.6 Выводы по разделу**

В ходе выполнения работы над разделом «Социальная ответственность» были выявлены опасные и вредные факторы, воздействию которых может подвергнуться работник, разрабатывающий алгоритм сегментации лёгких на снимках компьютерной томографии. Среди этих факторов: шум, уровень освещённости, показатели микроклимата, психофизиологические факторы и электрический ток. По результатам работы, эти показатели не превышают допустимых значений. Также был проведен анализ нормативной документации.

Основываясь на результатах проделанной работы, был предложен ряд мер, для исключения или уменьшения влияния опасных и вредных факторов на человека и окружающую среду. Таким образом, рабочее место соответствует нормативным требованиям по безопасности.

## **Заключение**

В ходе данной работы был рассмотрен датасет со снимками легких КТ и предложено использование модели глубокого обучения Unet с энкодерами на основе архитектур ResNet-34, MobileNetV2 и EfficientNet-B0, предобученных на датасете ImageNet. Для расширения обучающей выборки были использованы аугментации сдвига, вращения вокруг центра, увеличения, и отражения вокруг горизонтальной оси. На языке программирования Python с использованием фреймворка Tensorflow, библиотек scikit-learn и segmentation-models было реализовано программное обеспечение для подготовки и аугментации данных, обучения моделей и их проверки на обучающей выборке. После обучения модели были проверена на тестовой выборке, лучший результат показала модель на основе архитектуры EfficientNet-B0 – метрика IoU составила 0.9722. Также данная архитектура является оптимальной по точности и размеру модели.

## **Список публикаций и научных достижений**

### Участие в конференциях:

1. Диплом II степени за лучший доклад на подсекции 3.4 «Вычислительный интеллект» Международной научно-технической конференции студентов, аспирантов и молодых ученых «НАУЧНАЯ СЕССИЯ ТУСУР», г. Томск, 25-27 мая 2020 г.

### Участие в конкурсах:

1. Диплом I степени на конкурсе по решению бизнес-кейсов в сфере медицины "МедХакатон", г. Томск, сентябрь 2018 г.;
2. Диплом за 3 место на II отборочных соревнованиях Digital Skills Томской области, г. Томск, май 2019 г.

### Публикации:

1. Zarnitsyn A. Y. et al. Development of the video stream object detection algorithm (VSODA) with tracking // EAI Endorsed Transactions on Energy Web. – 2019. – Т. 6. – №. 22.

## Список используемых источников

1. Osborne J. Google's tensor processing unit explained: this is what the future of computing looks like // TechRadar. Available via <http://www.techradar.com/>. Accessed. – 2017. – Т. 6.
2. Ivakhnenko A. G., Lapa V. G. Cybernetics and forecasting techniques. – 1967.
3. Fukushima K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position // Biological cybernetics. – 1980. – Т. 36. – №. 4. – С. 193-202.
4. LeCun Y. et al. Backpropagation applied to handwritten zip code recognition // Neural computation. – 1989. – Т. 1. – №. 4. – С. 541-551.
5. Krizhevsky A., Sutskever I., Hinton G. E. Imagenet classification with deep convolutional neural networks // Advances in neural information processing systems. – 2012. – С. 1097-1105.
6. Goodfellow I., Bengio Y., Courville A. Deep learning. – MIT press, 2016.
7. Collobert R., Bengio S. Links between perceptrons, MLPs and SVMs // Proceedings of the twenty-first international conference on Machine learning. – 2004. – С. 23.
8. Srivastava N. et al. Dropout: a simple way to prevent neural networks from overfitting // The journal of machine learning research. – 2014. – Т. 15. – №. 1. – С. 1929-1958.
9. Hinton G. E. et al. Improving neural networks by preventing co-adaptation of feature detectors // arXiv preprint arXiv:1207.0580. – 2012.
10. Николенко С., Кадурин А., Архангельская Е. Глубокое обучение. – "Издательский дом " Питер"", 2017.
11. Ronneberger O., Fischer P., Brox T. U-net: Convolutional networks for biomedical image segmentation // International Conference on Medical

- image computing and computer-assisted intervention. - Springer, Cham, 2015. - C. 234-241.
12. Iglovikov V., Shvets A. Terausnet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation // arXiv preprint arXiv:1801.05746. - 2018.
  13. Deng J. et al. Imagenet: A large-scale hierarchical image database // 2009 IEEE conference on computer vision and pattern recognition. - Ieee, 2009. - C. 248-255.
  14. He K. et al. Deep residual learning for image recognition // Proceedings of the IEEE conference on computer vision and pattern recognition. - 2016. - C. 770-778.
  15. Simonyan K., Zisserman A. Very deep convolutional networks for large-scale image recognition // arXiv preprint arXiv:1409.1556. - 2014.
  16. Sandler M. et al. Mobilenetv2: Inverted residuals and linear bottlenecks // Proceedings of the IEEE conference on computer vision and pattern recognition. - 2018. - C. 4510-4520.
  17. Tan M., Le Q. V. Efficientnet: Rethinking model scaling for convolutional neural networks // arXiv preprint arXiv:1905.11946. - 2019.
  18. Martín Abadi et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Режим доступа: <https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf> (Дата доступа 31.05.2020)
  19. Pavel Yakubovski. Segmentation Models, 2019. Режим доступа: [https://github.com/qubvel/segmentation\\_models](https://github.com/qubvel/segmentation_models) (Дата доступа 31.05.2020)
  20. Kevin Mader. Finding and Measuring Lungs in CT Data // [Электронный ресурс]. Режим доступа: <https://www.kaggle.com/kmader/finding-lungs-in-ct-data> (Дата доступа 31.05.2020)

21. Трудовой кодекс Российской Федерации от 30 декабря 2001 г. № 197-ФЗ (с изм. от 1 апреля 2019 г.) // Собрание законодательства РФ. 2002. № 1 (ч. 1). Ст. 3.
22. ГОСТ "Система стандартов безопасности труда (ССБТ). Рабочее место при выполнении работ сидя. Общие эргономические требования." №12.2.032-78.
23. ГОСТ ССБТ. "Оборудование производственное. Общие требования безопасности к рабочим местам". №12.2.061-81.
24. СанПиН "Гигиенические требования к персональным электронно-вычислительным машинам и организации работы." от 21 июня 2003 г. №2.2.2/2.4.1340-03.
25. ГОСТ "Система стандартов безопасности труда (ССБТ). Опасные и вредные производственные факторы. Классификация." №12.0.003-2015.
26. СанПиН 2.2.4.548-96. Гигиенические требования к микроклимату производственных помещений.
27. СН 2.2.4/2.1.8.562-96. Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки.
28. СП 52.13330.2016 Естественное и искусственное освещение. Актуализированная редакция СНиП 23-05-95\*.
29. ГОСТ Р 12.1.019-2009. Электробезопасность. Общие требования и номенклатура видов защиты.
30. СанПиН 2.2.1/2.1.1.1278-03. Гигиенические требования к естественному, искусственному и совмещенному освещению жилых и общественных зданий.
31. ГОСТ 12.1.030-81. Электробезопасность. Защитное заземление. Зануление.
32. ГОСТ 30775-2001 Ресурсосбережение. Обращение с отходами. Классификация, идентификация и кодирование отходов.

33. ГОСТ 12.1.004-91. Пожарная безопасность. Общие требования.
34. ГОСТ 12.2.037-78. Техника пожарная. Требования безопасности.
35. СНиП 21-01-97. Противопожарные нормы.
36. НПБ 105-03. Определение категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности.
37. ГОСТ 12.4.154. Система стандартов безопасности труда. Устройства экранирующие для защиты от электрических полей промышленной частоты. Общие технические требования, основные параметры и размеры
38. СНиП 23-05-95 "Естественное и искусственное освещение"
39. Пособие к МГСН 2.06-99. Расчет и проектирование искусственного освещения помещений общественных зданий. Москомархитектура. 1999 г.
40. СН "Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории застройки." от 31 октября 1996 г. № 2.2.4/2.1.8.562 – 96.
41. ГОСТ "Здания жилые и общественные. Параметры микроклимата в помещениях." от 1 января 2013 г. № 30494—2011.
42. ГОСТ Р "Безопасность оборудования информационной технологии, включая электрическое конторское оборудование." № 50377-92 (МЭК 950-86).

**Приложение А**  
(справочное)

**Deep learning for semantic segmentation tasks**

Студент:

Группа	ФИО	Подпись	Дата
8ПМ8И	Войцеховский Алексей Алексеевич		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Аксёнов Сергей Владимирович	к.т.н.		

Консультант-лингвист ОИЯ ШБИП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Пичугова Инна Леонидовна			

## 1.1 Deep learning

Deep learning is a combination of machine learning methods based on representation training. Deep learning is currently the state-of-the-art approach in solving the problems of computer vision, natural language processing, speech recognition, etc. The success of deep learning is associated with the development of computers, including graphic and specialized tensor processor units [1].

Deep learning has become widespread recently, but attempts to train algorithms based on multilayer neural networks have been carried out since the 60s of the 20th century [2]. In 1980, Kunihiko Fukushima introduced the Neocognitron model, designed to recognize patterns on the images [3].

In 1989, Yann LeCun et al. proposed using the backpropagation algorithm for deep convolution networks training. They trained a convolutional neural network that could recognize ZIP codes. The training dataset included 9298 images of digits prepared on the basis of handwritten ZIP codes. The algorithm was trained 3 days. The final model was implemented on a digital signal processor and could process 10 digits per second [4].

Then a lot of work was done to develop the methodology for applying deep learning. However, a breakthrough occurred only in 2012, when Alex Krizhevsky, Ilya Sutskever and Jeffrey Hinton from the University of Toronto won the ImageNet image classification contest and published an article [5]. Before, the ImageNet was won by classic computer vision algorithms that used hand-crafted features. The authors used regularization with the dropout method, augmentation of the training set, and other deep learning techniques for training AlexNet model. The authors also used graphic processor units to accelerate neural networks. Parallelization of neural network computing has led to the widespread use of deep learning in the academic and in the commercial fields.

The success of deep learning was also influenced by the availability of a large amount of data. Figure 1 shows increasing the size of the datasets over time [6].

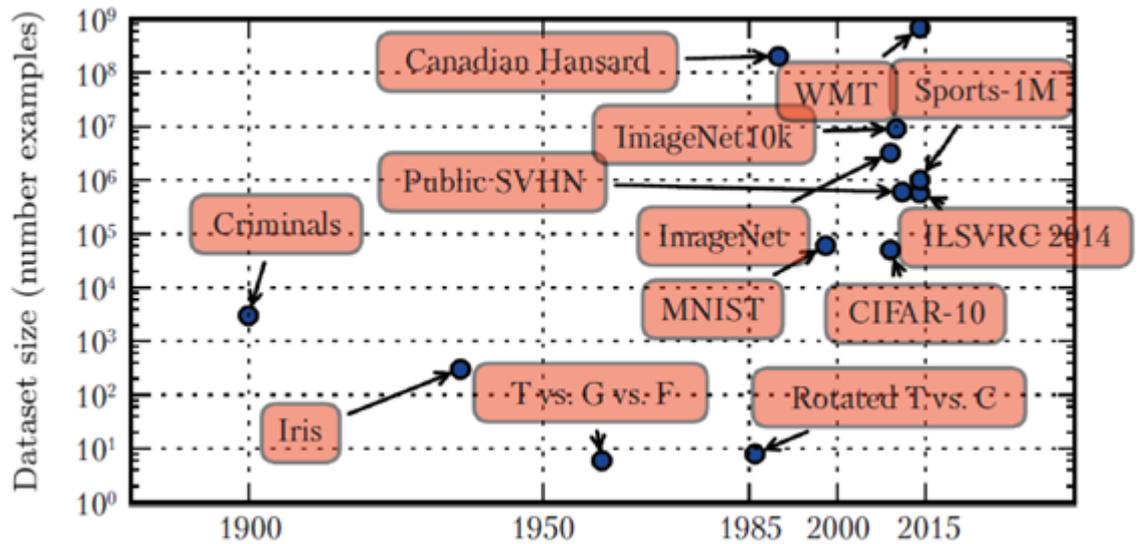


Figure 1. Increasing the size of the datasets over time

In the early 1900s, statisticians studied datasets containing from hundreds to thousands of manually prepared measurements. Between the 1950s and 1980s, computer vision pioneers often worked with small synthetic data sets, such as low-resolution bitmap images of letters that were specifically designed to reduce the cost of computing and demonstrate that neural networks could be trained in special-purpose functions. In the 1980s and 1990s, machine learning became more statistical, and datasets already totaled tens of thousands of examples, such as the set of scanned handwritten digits MNIST. In the first decade of the 21st century, they continued to create more sophisticated data sets of the same size, such as CIFAR-10. At the end of this period and in the first half of the 2010s, the advent of large data sets containing from hundreds of thousands to tens of millions of examples completely changed the concept of deep learning opportunities. These datasets include the Street View House Numbers, a wide variety of ImageNet dataset, and Sports-1M dataset. In the upper part of the diagram, we see that the datasets of translated sentences, for example, the dataset of the official records of the meetings of the Canadian Parliament collected by IBM, and the dataset of WMT 2014 translations from English to French, are much larger in size than most other sets [6].

## 1.2 Neural networks overfitting and methods for preventing overfitting

One of the challenges for deep learning methods is overfitting. In machine learning, overfitting is the phenomenon when a trained model does not work well with examples that were not in the training dataset. Overfitting occurs when a model is trained on hidden random patterns in training data, but not in the general population. Overfitting characterizes the generalizing ability of the model.

The following regularization techniques are used for solving the problem of overfitting:

1. L1- and L2-regularization of weights
2. Early stopping
3. The dropout method

The meaning of the first technique is limiting the absolute values of the weights. For this, one of the regularizers is applied to the loss function:

3. L1-regularizer – the sum of the squares of the weights  $L1 = \lambda \sum_{\omega} \omega^2$
4. L2-regularizer – the sum of the modules of the weights  $L2 = \lambda \sum_{\omega} |\omega|$

The second technique is quite simple. We should put part of the training data into the validation data. When the error begins to grow, it means that the neural network has begun to overfit and it is necessary to stop the training process. There is evidence that this technique is similar to L2-regularization [7]. Moreover, error analysis on the validation dataset allows us to evaluate the generalized ability of models.

The third method is more effective than the first. The idea of the “dropout” method is that for each neuron (except for the output layer), the probability  $p$  is established with which it will be ejected from the neural network. The training algorithm changes in this way: on each new training example  $x$ , we first determine for each neuron activity with probability  $p$ , and depending on the result we either use the neuron as usual, or set its output to always be strictly equal to zero. Then everything happens without changes. Zero at the output leads to the fact that the neuron actually

drops out of the computation graph: both direct computation and the reverse propagation of the gradient stop at this neuron and do not go further [8]. The experiments [9] and the practice of training neural networks show that the “dropout” method does provide very serious improvements in the quality of the trained model in a wide variety of applications.

### **1.3 Convolutional neural networks**

Usually convolutional neural networks are used to solve computer vision problems. The Neocognitron is considered to be the initially implemented convolutional neural network, although this algorithm does not use teaching methods with a teacher [3]. It is commonly believed that convolutional neural networks simulate the functioning of the visual cortex [10]. It is divided into zones called visual area one (V1) and so on until visual area seven (V7). Visual areas differ in structure, physiology and function [10]. Modern neurophysiology distinguishes the following functions:

- 11.V1 extracts local features of small areas of the image from the retina;
- 12.V2 also extracts local features and generalizes them, and it is also responsible for binocular vision;
- 13.V3 recognizes the colors and textures of objects, and also segments them;
- 14.V4 recognizes geometric shapes and is responsible for focusing on a specific subject;
- 15.V5 recognizes the direction and speed of objects;
- 16.V6 summarizes the whole picture;
- 17.V7 recognizes complex objects, including human faces.

Deep convolutional neural networks have a similar structure as visual cortex: the first layers extract simple features – edges and colors. Subsequent layers based on these features extract more complex features, such as shapes. The last layers are responsible for object recognition [10].

The main idea of convolutional neural networks: the processing of image part should not depend on the specific location of this part. Therefore, we cover the input image with small windows (for example, 5x5 pixels) and extract features in each window with a small neural network. It will have only  $5 \times 5 = 25$  inputs, and from each image it can get a lot of different inputs [6]. Then the results can again be presented in the form of an image, replacing the  $5 \times 5$  windows with their central pixels. Then we can apply a second convolutional layer with another small neural network to this image. In each convolutional layer there are very few free parameters, especially in comparison with fully connected neural networks [6].

Typically, the original image consists of three matrices (by RGB color channels) – red, green and blue. These matrices are called channels. The same matrices will be obtained after each convolutional layer. They have a spatial structure corresponding to the original picture. There can now be more channels. The values of each feature that we extracted from the windows in the original image will now be a whole matrix. Each matrix is called a feature map [6].

Next, we will consider how convolutional layers work. A convolution is a special linear transformation of input data. If  $x^l$  is the feature map in the layer with number  $l$ , then the result of 2d-convolution with a kernel of size  $2d + 1$  and a weight matrix  $W$  of size  $(2d + 1) \times (2d + 1)$  on the next layer will be like this [10]:

$$y_{i,j}^l = \sum_{-d \leq a, b \leq d} W_{a,b} x_{i+a, j+b}^l$$

where  $y_{i,j}^l$  – result of convolution on layer  $l$ ,  $x_{i,j}^l$  – its input, i.e. the output of the entire previous layer. We apply a linear transformation to the square window of the previous level to obtain a component of the next level, that is, we scalarly multiply the pixels from the window by the convolution vector.

The convolution operation has the following properties:

18. The convolution saves the structure of the input image, as it is applied to each section of the input data separately;

19. The convolution operation has the sparse property, since the value of each neuron of the next layer depends only on a small fraction of the input neurons;
20. The convolution uses the same weights many times, as they are reapplied to different parts of the entrance.

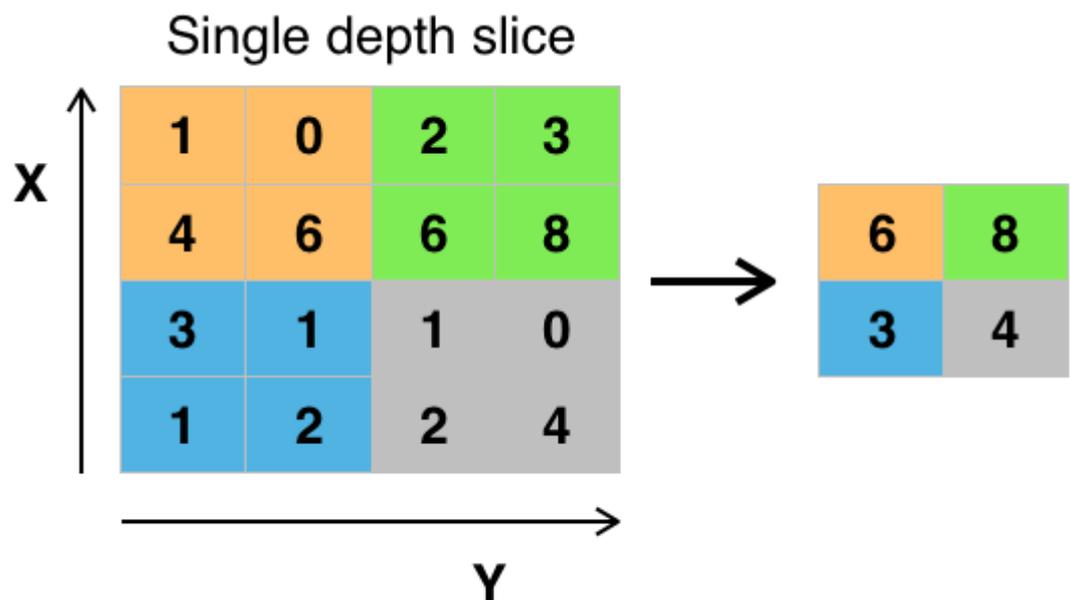
We can apply the nonlinear function  $h$  after the convolution operation [4]. It will simply be applied to each element of the resulting tensor individually. In the classical convolutional layer, in addition to linear convolution and non-linearity following it, there is one more operation: pooling.

In convolutional networks, the fact of the presence or absence of one or another feature is usually more important than its exact coordinates. For example, in the case of face recognition, it is more important to understand whether the image has a face and whose face it is than to find out where it is located. Therefore, we can generalize the extracted features, having lost some of the information about their location, but at the same time reducing the dimension [6].

Usually max-pooling operation is applied to each local group of neurons. The formula for calculating pooling is as follows [10]:

$$x_{i,j}^{l+1} = \max_{-d \leq a \leq d, -d \leq b \leq d} z_{i+a, j+b}^l$$

An example of the operation of the pooling layer is shown in Figure 2.



## Figure 2. Max-pooling

Another important element is the activation function, which was mentioned above. The activation function after the convolution layer is a source of non-linearity, which allows us to train neural networks. The activation function must meet the following criteria:

4. To be monotonously non-decreasing;
5. To be limited, i.e. tend to zero at  $x \rightarrow -\infty$ ;
6. To be differentiable.

There are many activation functions: logistic sigmoid, hyperbolic tangent, and others. However, rectified linear unit (ReLU) is more popular function for convolutional neural networks [10]. It has the following formula [10]:

$$ReLU(x) = \begin{cases} 0, & \text{if } x < 0, \\ x, & \text{if } x \geq 0. \end{cases}$$

ReLU is more effective than logistic sigmoid and hyperbolic tangent, since complex calculations are not needed to calculate the derivative of this function: if  $x < 0$ , then the derivative is zero, if  $x \geq 0$ , then the derivative is unity. It also allows you to reduce the cost of computing resources. In addition to the classic ReLU implementation, there are modifications of it that allow you to get the best result in some tasks [6].

### 1.4 Semantic segmentation architectures

In this paper, we solve the problem of semantic segmentation. Semantic segmentation in computer vision is understood as assigning a label of a certain class to each image pixel. For example, in the case of the lung segmentation task, each pixel must be assigned to one of two classes – lung or background.

U-net architecture is usually used as a model for semantic segmentation [11]. U-net was introduced by Olaf Ronnenberg et al. in 2015. Then the available architectures for image segmentation required training datasets consisting of thousands of images. The author was solving problems of processing biomedical

images and proposed this architecture to solve problem of lack of data. It allowed achieving good results even on small datasets. Furthermore, U-net is a fairly lightweight architecture and does not require significant computing resources. It is also universal and allows solving the problems of segmentation of both biomedical images and images from other areas, for example, the problem of segmentation of objects in satellite images [12].

Let us consider architecture of U-net in more detail. The U-net scheme is shown in Figure 3 [11].

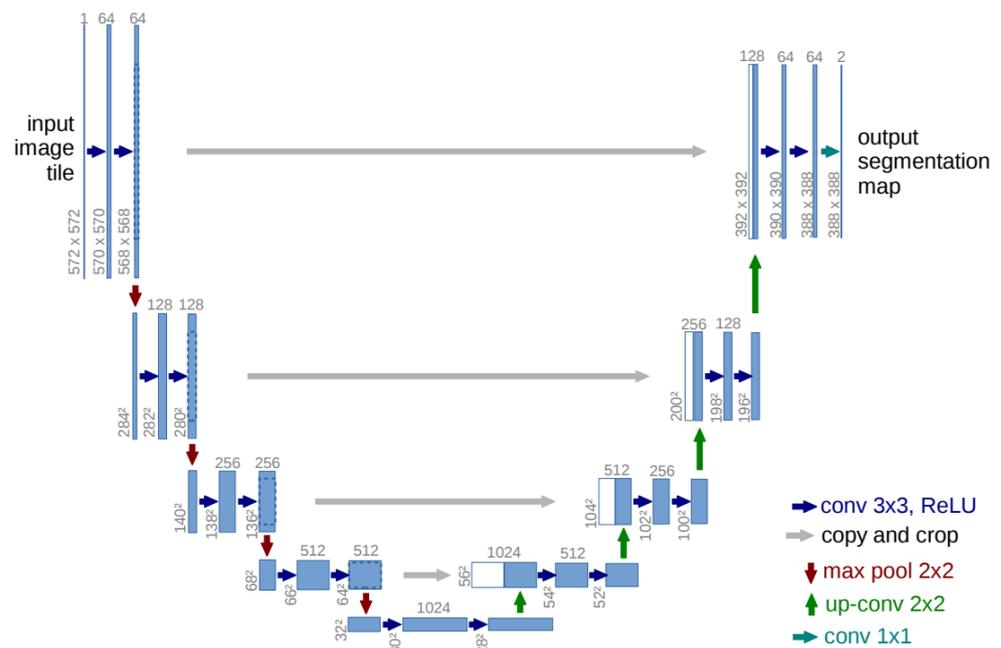


Figure 3. General scheme of the U-net architecture

U-net has a contracting path on the left side and an expansive path on the right side. The contracting path is a typical convolutional neural network architecture. It contains convolution blocks, ReLU activation functions, and maximum pooling operations. The expansive path also consists of convolution blocks and the ReLU activation function, but instead of pooling blocks, the up-convolutional blocks are used. Also, feature maps before each combining unit are transferred to the corresponding layers of the expansive path and are concatenated. In this case, the edges are padded, because convolution leads to the loss of pixels at the edges of the image. The last layer is a 1x1 convolution and is designed to display 64 feature cards

in the output image with the required number of classes. In this work, 2 classes are required – lung and background.

## **1.5 Encoders architectures**

The classic U-net contains 23 layers. Though, the modified U-net architecture is widely used to solve many tasks. Here, the contracting path is replaced by some common convolutional architecture, usually pre-trained on a dataset from a close domain or on a dataset with a large number of classes and examples, for instance, ImageNet [13]. This often allows us to increase the accuracy of segmentation by increasing the number of layers and the use of other techniques to improve the quality of the model [12]. In this work, the following architectures were used as encoders for U-net:

1. ResNet-34 [14];
2. MobileNetV2 [16];
3. EfficientNet-B0 [17].

Next, we consider them in more detail.

### **1.5.1 ResNet**

The ResNet architecture family was introduced in 2015 by a group of researchers from Microsoft [14]. The deeper the neural network is, the more difficult it is to train. Therefore, Kaiming He et al. have proposed an approach called residual neural networks. The main idea of this approach is to add residual connections between groups of layers, forming a residual block, as shown in Figure 4 [14]. Such units allow reaching a depth of the neural network of 152 layers, which is 8 times more than the VGG networks [15]. At the same time, ResNet retain less computational complexity [14]. As a result, this approach makes it relatively easy to increase the accuracy of the neural network while increasing the number of layers.

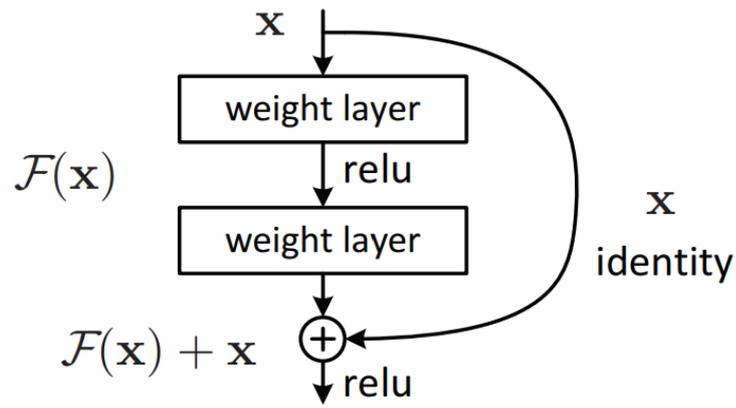


Figure 4. Residual block in ResNet

### 1.5.2 MobileNetV2

MobileNetV2 architecture was introduced in 2019 by Mark Sandler et al. from Google [16]. The aim of this work was to achieve a new level of quality on mobile platforms, such as smartphones. The task is complicated by the requirements for the size and energy efficiency of mobile processors. This architecture is based on the previously discussed ResNet. A feature of this architecture is convolutional blocks with a bottleneck that does not contain nonlinearities. Figures 5 and 6 show a common convolution block and a bottle-neck convolution block (the hatched layer does not contain nonlinearities) [16].

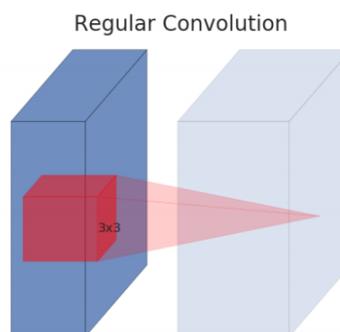


Figure 5. Common convolution unit

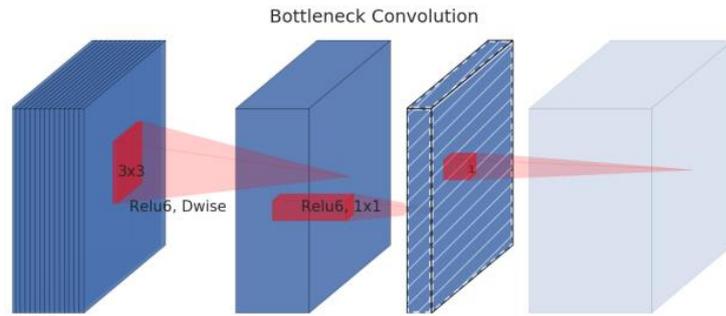


Figure 6. Bottleneck convolution unit

The second feature is the reverse residual blocks instead of residual blocks. Their difference is that the residual connections connect the bottleneck blocks. In general, this architecture allows us to find a compromise between accuracy and performance, which can be useful if the model needs to be used on weak personal computers or mobile devices.

### 1.5.3 EfficientNet

The EfficientNet family of architectures was introduced in 2019 by Mingxing Tan and Quoc V. Le from Google [17]. The authors used the neural architecture search approach to search for new architectures. Using this method, varying the depth and width of the neural network, and image resolution, they received a new family of neural networks EfficientNet. New neural networks showed better results than existing architectures, with a smaller size and computational complexity [17].

### 1.6 Transfer learning

As it was mentioned earlier, transfer learning of neural networks is widely used in solving segmentation problems. This is due to an improvement in the process of convergence of neural networks compared with random initialization of weights [12]. The most common dataset for pre-training is ImageNet [13]. This choice is not accidental, because ImageNet is the largest database of annotated images and

contains more than 1000 classes, which allows us to train a neural network for most domains.

## **1.7 Data augmentation**

Training dataset augmentation is a common practice in machine learning. Augmentation is the extending of the training dataset by generating new examples based on existing ones using various distortions. For images, shifts, rotations, scaling, distortions, color channel shifts, etc. are usually used. For example, when working on the classical U-net architecture, the authors used various augmentations to improve the quality of the model [11]. Another example is the article by Vladimir Iglovikov about using of modified U-net for image segmentation in satellite images, which discusses the improvement of the model learning process due to data augmentation [12].

## References

1. Osborne J. Google's tensor processing unit explained: this is what the future of computing looks like // TechRadar. Available via <http://www.techradar.com/>. Accessed. – 2017. – Т. 6.
2. Ivakhnenko A. G., Lapa V. G. Cybernetics and forecasting techniques. – 1967.
3. Fukushima K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position // Biological cybernetics. – 1980. – Т. 36. – №. 4. – С. 193-202.
4. LeCun Y. et al. Backpropagation applied to handwritten zip code recognition // Neural computation. – 1989. – Т. 1. – №. 4. – С. 541-551.
5. Krizhevsky A., Sutskever I., Hinton G. E. Imagenet classification with deep convolutional neural networks // Advances in neural information processing systems. – 2012. – С. 1097-1105.
6. Goodfellow I., Bengio Y., Courville A. Deep learning. – MIT press, 2016.
7. Collobert R., Bengio S. Links between perceptrons, MLPs and SVMs // Proceedings of the twenty-first international conference on Machine learning. – 2004. – С. 23.
8. Srivastava N. et al. Dropout: a simple way to prevent neural networks from overfitting // The journal of machine learning research. – 2014. – Т. 15. – №. 1. – С. 1929-1958.
9. Hinton G. E. et al. Improving neural networks by preventing co-adaptation of feature detectors // arXiv preprint arXiv:1207.0580. – 2012.
10. Николенко С., Кадуриин А., Архангельская Е. Глубокое обучение. – "Издательский дом " Питер"", 2017.
11. Ronneberger O., Fischer P., Brox T. U-net: Convolutional networks for biomedical image segmentation // International Conference on Medical image computing and computer-assisted intervention. - Springer, Cham, 2015. - С. 234-241.

12. Igloukov V., Shvets A. Ternaunet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation // arXiv preprint arXiv:1801.05746. - 2018.
13. Deng J. et al. Imagenet: A large-scale hierarchical image database // 2009 IEEE conference on computer vision and pattern recognition. - Ieee, 2009. - C. 248-255.
14. He K. et al. Deep residual learning for image recognition // Proceedings of the IEEE conference on computer vision and pattern recognition. - 2016. - C. 770-778.
15. Simonyan K., Zisserman A. Very deep convolutional networks for large-scale image recognition // arXiv preprint arXiv:1409.1556. - 2014.
16. Sandler M. et al. Mobilenetv2: Inverted residuals and linear bottlenecks // Proceedings of the IEEE conference on computer vision and pattern recognition. - 2018. - C. 4510-4520.
17. Tan M., Le Q. V. Efficientnet: Rethinking model scaling for convolutional neural networks // arXiv preprint arXiv:1905.11946. - 201

## Приложение Б. Исходный код процесса обучения и проверки моделей

```
import segmentation_models as sm
from tensorflow import keras
import os
import numpy as np
import pandas as pd
import cv2
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
%matplotlib inline

sm.set_framework('tf.keras')
PATH = 'drive/My Drive/'
IMAGE_LIB = f'{PATH}data/2d_images/'
MASK_LIB = f'{PATH}data/2d_masks/'
IMG_HEIGHT, IMG_WIDTH = 256, 256
SEED=42

all_images = [x for x in sorted(os.listdir(IMAGE_LIB)) if x[-4:] == '.tif']

x_data = np.empty((len(all_images), IMG_HEIGHT, IMG_WIDTH, 3), dtype='float32')
for i, name in enumerate(all_images):
    im = cv2.imread(IMAGE_LIB + name, cv2.IMREAD_UNCHANGED).astype("int16").astype('float32')
    im = cv2.resize(im, dsize=(IMG_WIDTH, IMG_HEIGHT), interpolation=cv2.INTER_LANCZOS4)
    im = (im - np.min(im)) / (np.max(im) - np.min(im))
    img = np.zeros((IMG_WIDTH, IMG_HEIGHT, 3))
    img[:, :, 0] = im
    img[:, :, 1] = im
    img[:, :, 2] = im
    x_data[i] = img

y_data = np.empty((len(all_images), IMG_HEIGHT, IMG_WIDTH), dtype='float32')
for i, name in enumerate(all_images):
    im = cv2.imread(MASK_LIB + name, cv2.IMREAD_UNCHANGED).astype('float32')/255.
    im = cv2.resize(im, dsize=(IMG_WIDTH, IMG_HEIGHT), interpolation=cv2.INTER_NEAREST)
    y_data[i] = im

x_data = x_data[:,:,:,:]
y_data = y_data[:,:,:,np.newaxis]
x_train, x_val, y_train, y_val = train_test_split(x_data, y_data, test_size = 0.5)
```

```

x_val, x_test, y_val, y_test = train_test_split(x_val, y_val, test_size = 0.5)
data_gen_args = dict(#featurewise_center=True,
                    #featurewise_std_normalization=True,
                    rotation_range=5,
                    width_shift_range=0.05,
                    height_shift_range=0.05,
                    zoom_range=0.2,
                    horizontal_flip=True)

image_datagen = keras.preprocessing.image.ImageDataGenerator(**data_gen_args)
mask_datagen = keras.preprocessing.image.ImageDataGenerator(**data_gen_args)

image_datagen_val = keras.preprocessing.image.ImageDataGenerator()
mask_datagen_val = keras.preprocessing.image.ImageDataGenerator()

image_datagen.fit(x_train, augment=True, seed=SEED)
mask_datagen.fit(y_train, augment=True, seed=SEED)
image_datagen_val.fit(x_val, augment=False, seed=SEED)
mask_datagen_val.fit(y_val, augment=False, seed=SEED)

image_generator = image_datagen.flow(
    x_train,
    batch_size=16,
    seed=SEED)
mask_generator = mask_datagen.flow(
    y_train,
    batch_size=16,
    seed=SEED)

image_generator_val = image_datagen_val.flow(
    x_val,
    seed=SEED)
mask_generator_val = mask_datagen_val.flow(
    y_val,
    seed=SEED)

train_generator = (pair for pair in zip(image_generator, mask_generator))

val_generator = (pair for pair in zip(image_generator_val, mask_generator_val))

def model_callbacks(model_name):
    callbacks = [
        keras.callbacks.ModelCheckpoint(
            filepath=model_name + '/' + 'mymodel_{epoch}',

```

```

        # Path where to save the model
        # The two parameters below mean that we will overwrite
e
        # the current checkpoint if and only if
        # the `val_loss` score has improved.
        save_best_only=True,
        monitor='val_loss',
        verbose=1),
        keras.callbacks.CSVLogger(f'{model_name}/training.log'),
        #keras.callbacks.TensorBoard(log_dir='./logs'), # How of
ten to write logs (default: once per epoch)
        keras.callbacks.EarlyStopping(monitor='val_loss', min_delta=0,
patience=10, verbose=0, mode='auto', baseline=None, restore_best_weights=True)
    ]
    return callbacks

```

```

ARCHS = ('unet',)
BACKBONES = ('resnet34', 'mobilenetv2', 'efficientnetb0')
for arch in ARCHS:
    for backbone in BACKBONES:
        preprocess_input = sm.get_preprocessing(backbone)
        model_path = f'{PATH}data/{arch}/{backbone}'
        # load your data

        # preprocess input
        X_TRAIN = x_train
        X_VAL = x_val
        X_TEST = x_test
        #X_TRAIN = preprocess_input(x_train)
        #X_VAL = preprocess_input(x_val)
        #X_TEST = preprocess_input(x_test)

        # define model
        if arch == 'unet':
            model = sm.Unet(backbone, encoder_weights='imagenet')
        else:
            model = sm.FPN(backbone, encoder_weights='imagenet')
        model.compile(
            'Adam',
            loss=sm.losses.bce_jaccard_loss,
            metrics=[sm.metrics.iou_score],
        )
        callbacks = model_callbacks(model_path)

        model.fit(
            train_generator,

```

```

        steps_per_epoch=20,
        epochs=100,
        validation_data=val_generator,
        validation_steps=len(X_VAL),
        callbacks=callbacks
    )

    score = model.evaluate(X_TEST, y_test)
    with open(model_path + '/' + 'score.txt', 'w') as f:
        for i in score:
            f.write(str(i) + '\n')
    prediction = model.predict(X_TEST)
    idx = 0
    for img, gt, pred in zip(x_test, y_test, prediction):
        fig, ax = plt.subplots(1, 4, figsize = (16,8))
        ax[0].imshow(img, cmap='gray')
        ax[0].set_title("a")
        ax[1].imshow(gt.reshape((256, 256)))
        ax[1].set_title("G")
        ax[2].imshow(pred.reshape((256, 256)))
        ax[2].set_title("P")
        ax[3].imshow(gt.reshape((256, 256)) -
pred.reshape((256, 256)))
        ax[3].set_title("r")
        plt.savefig(model_path + '/' + str(idx) + '.png')
        idx += 1

```