



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

Школа – Инженерная школа информационных технологий и робототехники  
Направление подготовки – 09.04.01 Информатика и вычислительная техника  
ООП/ОПОП – Data Science in Software Engineering / Анализ данных в разработке программного обеспечения  
Отделение школы (НОЦ) – Отделение информационных технологий

### **ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА МАГИСТРАНТА**

Тема работы
<b>Сервис обнаружения болезней сельскохозяйственных культур и насекомых-вредителей на основе аналитики больших данных</b>

УДК 004.65:632

Обучающийся

Группа	ФИО	Подпись	Дата
8ВМ1И	Жиганов Владислав Вадимович		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент (ОИТ ИШИТР)	Ковин Р. В.	К.Т.Н., доцент		

### **КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:**

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент (ОСГН, ШБИП)	Спицына Л. Ю.	К.Э.Н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент (ООД, ШБИП)	Антоневич О. А.	к.б.н.		

### **ДОПУСТИТЬ К ЗАЩИТЕ:**

Руководитель ООП, должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент (ОИТ, ИШИТР)	Зарубин А. Г.	К.Х.Н.		

**ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ООП**  
по направлению 09.04.01 Информатика и вычислительная техника

<b>Код компетенции</b>	<b>Наименование компетенции</b>
<b>Универсальные компетенции</b>	
УК(У)-1	Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий
УК(У)-2	Способен управлять проектом на всех этапах его жизненного цикла
УК(У)-3	Способен организовывать и руководить работой команды, вырабатывая командную стратегию для достижения поставленной цели
УК(У)-4	Способен применять современные коммуникативные технологии, в том числе на иностранном (-ых) языке (-ах), для академического и профессионального взаимодействия
УК(У)-5	Способен анализировать и учитывать разнообразие культур в процессе межкультурного взаимодействия
УК(У)-6	Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки
<b>Общепрофессиональные компетенции</b>	
ОПК(У)-1	Способен самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте
ОПК(У)-2	Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач
ОПК(У)-3	Способен анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями
ОПК(У)-4	Способен применять на практике новые научные принципы и методы исследований
ОПК(У)-5	Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем
ОПК(У)-6	Способен разрабатывать компоненты программно-аппаратных комплексов обработки информации и автоматизированного проектирования
ОПК(У)-7	Способен адаптировать зарубежные комплексы обработки информации и автоматизированного проектирования к нуждам отечественных предприятий
ОПК(У)-8	Способен осуществлять эффективное управление разработкой программных средств и проектов

Профессиональные компетенции	
ПК(У)-1	Способен разрабатывать и администрировать системы управления базами данных
ПК(У)-2	Способен к созданию программного обеспечения для анализа, распознавания и обработки информации, систем цифровой обработки сигналов
ПК(У)-3	Способен управлять процессами и проектами по созданию (модификации) информационных ресурсов
ПК(У)-4	Способен осуществлять руководство разработкой комплексных проектов на всех стадиях и этапах выполнения работ
ПК(У)-5	Способен проектировать и организовывать учебный процесс по образовательным программам с использованием современных образовательных технологий



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

Школа – Инженерная школа информационных технологий и робототехники  
Направление подготовки – 09.04.01 Информатика и вычислительная техника  
ООП/ОПОП – Data Science in Software Engineering / Анализ данных в разработке программного обеспечения  
Отделение школы (НОЦ) – Отделение информационных технологий

УТВЕРЖДАЮ:  
Руководитель ООП

\_\_\_\_\_ А.Г. Зарубин  
(Подпись) (Дата) (Ф.И.О.)

**ЗАДАНИЕ**  
**на выполнение выпускной квалификационной работы**

В форме:

магистерской диссертации

Студенту:

Группа	ФИО
8ВМ1И	Жиганову Владиславу Вадимовичу

Тема работы:

**Сервис обнаружения болезней сельскохозяйственных культур и насекомых-вредителей на основе аналитики больших данных**

Утверждена приказом директора (дата, номер)

№ 138-44/с от 18.05.2023

Срок сдачи студентом выполненной работы:

27.06.2023

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ:**

**Исходные данные к работе**

*(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.).*

1. Набор больших данных, содержащий RGB-изображения сельскохозяйственных культур, пораженных заболеваниями
2. Использование свёрточных нейронных сетей (СНС) архитектуры EfficientNet.
3. Точность обнаружения болезней у разработанного алгоритма должна быть выше (97%), чем у большинства конкурентных решений

<p><b>Перечень подлежащих исследованию, проектированию и разработке вопросов</b></p> <p><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>	<ol style="list-style-type: none"> <li>1. Анализ моделей нейронных сетей, с целью разработки сервиса.</li> <li>2. Подготовка набора больших данных.</li> <li>3. Разработка, обучение, валидация и исследование модели нейронной сети архитектуры EfficientNet.</li> <li>4. Разработка сервиса на основе обученной модели.</li> <li>5. Финансовый менеджмент.</li> <li>6. Социальная ответственность.</li> <li>7. Раздел на английском языке.</li> </ol>
<p><b>Перечень графического материала</b></p> <p><i>(с точным указанием обязательных чертежей)</i></p>	<ol style="list-style-type: none"> <li>1. Архитектура модели EfficientNet.</li> <li>2. Таблицы результатов обучения, валидации и исследования модели архитектуры EfficientNet.</li> <li>3. Диаграмма Ганта.</li> </ol>
<p><b>Консультанты по разделам выпускной квалификационной работы</b></p> <p><i>(с указанием разделов)</i></p>	
<p><b>Раздел</b></p>	<p><b>Консультант</b></p>
<p>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</p>	<p>Спицына Л. Ю., Доцент (ОСГН, ШБИП), к.э.н.</p>
<p>Социальная ответственность</p>	<p>Антоневич О. А., Доцент (ООД, ШБИП), к.б.н.</p>
<p>Английский язык</p>	<p>Ануфриева Т. Н., Старший преподаватель (ОИЯ, ШБИП)</p>
<p><b>Названия разделов, которые должны быть написаны на русском и иностранном языках:</b></p>	
<p>Chapter 1. Analysis of the subject area</p>	

<p><b>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</b></p>	<p>30.01.2023</p>
--	-------------------

**Задание выдал руководитель:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
<p>Доцент (ОИТ ИШИТР)</p>	<p>Ковин Р. В.</p>	<p>к.т.н., доцент</p>		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
<p>8ВМ1И</p>	<p>Жиганов Владислав Вадимович</p>		



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

Школа – Инженерная школа информационных технологий и робототехники  
Направление подготовки – 09.04.01 Информатика и вычислительная техника  
ООП/ОПОП – Data Science in Software Engineering / Анализ данных в разработке программного обеспечения

Отделение школы (НОЦ) – Отделение информационных технологий

Период выполнения – Весенний семестр 2022 /2023 учебного года

Форма представления работы:

Магистерская диссертация

(бакалаврская работа, дипломный проект/работа, магистерская диссертация)

### КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН выполнения выпускной квалификационной работы

Срок сдачи студентом выполненной работы:

01.06.2023

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
01.06.2023	Основная часть ВКР, включая раздел на английском языке	70
01.06.2023.	Раздел «Социальная ответственность»	15
01.06.2023	Раздел «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»	15

**СОСТАВИЛ:**

**Руководитель ВКР**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент (ОИТ ИШИТР)	Ковин Р. В.	к.т.н., доцент		

**СОГЛАСОВАНО:**

**Руководитель ООП**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент (ОИТ, ИШИТР)	Зарубин А. Г.	к.х.н.		

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

Студенту:

<b>Группа</b>	<b>ФИО</b>
8ВМ1И	Жиганову Владиславу Вадимовичу

<b>Школа</b>	<b>Инженерная школа информационных технологий и робототехники</b>	<b>Отделение школы (НОЦ)</b>	<b>ОИТ</b>
<b>Уровень образования</b>	магистратура	<b>Направление/специальность</b>	09.04.01 Информатика и вычислительная техника

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

1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	<i>Бюджет проекта – не более 840 511,27 руб., в т.ч. затраты по оплате труда – не более 605 750,06 руб</i>
2. <i>Нормы и нормативы расходования ресурсов</i>	<i>Значение показателя интегральной ресурсоэффективности – не менее 4,75 баллов из 5</i>
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	<i>Коэффициент отчислений на уплату во внебюджетные фонды 30%</i>

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

1. <i>Оценка коммерческого и инновационного потенциала НТИ</i>	<i>Провести предпроектный анализ</i>
2. <i>Разработка устава научно-технического проекта</i>	<i>Представить Устав научного проекта магистерской работы</i>
3. <i>Планирование процесса управления НТИ: структура и график проведения, бюджет, риски и организация закупок</i>	<i>Разработать план управления НТИ</i>
4. <i>Определение ресурсной, финансовой, экономической эффективности</i>	<i>Рассчитать сравнительную эффективность исследования</i>

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

1. <i>Сегментирование рынка</i>
2. <i>Оценка конкурентоспособности технических решений</i>
3. <i>Матрица SWOT</i>
4. <i>Иерархическая структура работ</i>
5. <i>График проведения и бюджет НТИ</i>
6. <i>Оценка сравнительной эффективности НТИ</i>

<b>Дата выдачи задания для раздела по линейному графику</b>	
---	--

**Задание выдал консультант:**

<b>Должность</b>	<b>ФИО</b>	<b>Ученая степень, звание</b>	<b>Подпись</b>	<b>Дата</b>
Доцент (ОСГН, ШБИП)	Спицына Любовь Юрьевна	К.Э.Н.		

**Задание принял к исполнению студент:**

<b>Группа</b>	<b>ФИО</b>	<b>Подпись</b>	<b>Дата</b>
8ВМ1И	Жиганов Владислав Вадимович		

## ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

<b>Группа</b>		<b>ФИО</b>	
8ВМ1И		Жиганову Владиславу Вадимовичу	
<b>Школа</b>	<b>Инженерная школа информационных технологий и робототехники</b>	<b>Отделение (НОЦ)</b>	<b>ОИТ</b>
<b>Уровень образования</b>	магистратура	<b>Направление/специальность</b>	09.04.01 Информатика и вычислительная техника

Тема ВКР:

Сервис обнаружения болезней сельскохозяйственных культур и насекомых-вредителей на основе аналитики больших данных

**Исходные данные к разделу «Социальная ответственность»:**

<p><b>Введение</b></p> <ul style="list-style-type: none"> <li>– Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика) и области его применения.</li> <li>– Описание рабочей зоны (рабочего места) при разработке проектного решения/при эксплуатации</li> </ul>	<p>Объект исследования: сверточная нейронная сеть архитектуры EfficientNet, которая используется для обнаружения болезней сельскохозяйственных культур на изображениях;</p> <p>Область применения: системы компьютерного зрения, машинное обучение;</p> <p>Рабочая зона: офис;</p> <p>Размеры помещения: 4×3,5 м.;</p> <p>Количество и наименование оборудования рабочей зоны: персональный компьютер, ЖК монитор, офисное кресло, центральное отопление; естественная вентиляция; искусственное и естественное освещение;</p> <p>Рабочие процессы, связанные с объектом исследования, осуществляющиеся в рабочей зоне: сбор и обработка данных, разработка архитектур сверточных нейронных сетей, обучение и исследование сверточных нейронных сетей;</p>
--	--

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

<p><b>1. Правовые и организационные вопросы обеспечения безопасности при разработке проектного решения:</b></p> <ul style="list-style-type: none"> <li>– специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства;</li> <li>– организационные мероприятия при компоновке рабочей зоны.</li> </ul>	<p>ГОСТ 12.2.032-78 Рабочее место при выполнении работ сидя;</p> <p>ГОСТ 22269-76 Рабочее место оператора. Взаимное расположение элементов рабочего места;</p> <p>ГОСТ Р 50923-96. Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения;</p> <p>СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания.</p>
<p><b>2. Производственная безопасность при разработке проектного решения:</b></p> <ul style="list-style-type: none"> <li>– Анализ выявленных вредных и опасных производственных факторов</li> <li>– Расчет уровня опасного или вредного производственного фактора</li> </ul>	<p><b>Опасные факторы:</b></p> <p>1. Производственные факторы, связанные с электромагнитными полями, неионизирующими ткани тела человека постоянного характера, связанного с повышенным образованием электростатических зарядов.</p> <p><b>Вредные факторы:</b></p> <p>1. Отсутствие или недостатки необходимого искусственного освещения.</p>



	<p>2. Отклонение показателей микроклимата.  3. Повышенный уровень шума.  4. Перенапряжение анализаторов, в том числе вызванное информационной нагрузкой.  5. Рабочая поза.</p> <p><b>Требуемые средства коллективной и индивидуальной защиты от выявленных факторов:</b> использование устройств защитного заземления, увеличение источников искусственного освещения, использование очков при работе с ЭВМ.</p> <p><b>Расчет:</b> расчет системы искусственного освещения.</p>
3. Экологическая безопасность при разработке проектного решения	<p><b>Воздействие на селитебную зону:</b> отсутствует.</p> <p><b>Воздействие на литосферу:</b> неправильная утилизации комплектующих ЭВМ; неправильная утилизации люминесцентных ламп.</p> <p><b>Воздействие на гидросферу:</b> отсутствует.</p> <p><b>Воздействие на атмосферу:</b> выброс вредных веществ в результате выработки электроэнергии, необходимой для работы ЭВМ; неправильная утилизации комплектующих ЭВМ.</p>
4. Безопасность в чрезвычайных ситуациях при разработке проектного решения	<p><b>Возможные ЧС:</b> отказ систем безопасности, пожар, выход из строя электрической системы, землетрясение, пожар.</p> <p><b>Наиболее типичная ЧС:</b> пожар</p>
Дата выдачи задания для раздела по линейному графику	

**Задание выдал консультант:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент (ООД, ШБИП)	Антоневич Ольга Алексеевна	к.б.н.		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8ВМ1И	Жиганов Владислав Вадимович		

## РЕФЕРАТ

Выпускная квалификационная работа 96 с., 15 рис., 23 табл., 30 источников, 2 прил.

Ключевые слова: обнаружение болезней сельскохозяйственных культур на изображениях, свёрточная нейронная сеть, набор больших данных, архитектура EfficientNet, машинное обучение.

Целью данной работы является создание сервиса, способного автоматически выявлять заболевания сельскохозяйственных культур на изображениях.

Разработка сервиса обнаружения болезней сельскохозяйственных культур представляет собой создание программного продукта, который позволит определять наличие заболеваний у растений на основе анализа изображений с использованием архитектуры нейронных сетей EfficientNet. Для обучения моделей СНС используется набор данных, содержащий изображения здоровых и больных растений сельскохозяйственных культур. Результатом вычисления модели нейронной сети для каждого изображения является определение наличия или отсутствия заболевания и его типа.

Разработанный сервис может быть использован на предприятиях сельского хозяйства, занимающихся полеводством, овощеводством и садоводством, для раннего выявления заболеваний.

# ОГЛАВЛЕНИЕ

ОСНОВНЫЕ СОКРАЩЕНИЯ И ОБОЗНАЧЕНИЯ .....	15
ВВЕДЕНИЕ.....	16
ГЛАВА 1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ .....	17
1.1. Описание предметной области.....	17
1.2. Обзор существующих решений .....	18
1.2.1. ResNet-50 with SVM .....	18
1.2.2. GoogLeNet.....	18
1.2.3. Inception v3 transferred to target domain SVM.....	19
1.3. Вывод по анализу предметной области.....	19
ГЛАВА 2. РАЗРАБОТКА МЕТОДА ОБНАРУЖЕНИЯ БОЛЕЗНЕЙ РАСТЕНИЙ .....	21
2.1. Описание используемой архитектуры нейронной сети.....	21
2.1.1. Метод составного масштабирования.....	21
2.1.2. Архитектура модели EfficientNet .....	22
2.1.3. Производительность модели EfficientNet .....	23
2.2. Используемый набор данных .....	24
2.3. Обучение модели .....	26
2.4. Оценка показателей производительности разработанной модели 27	
ГЛАВА 3. РАЗРАБОТКА СЕРВИСА НА ОСНОВЕ ОБУЧЕННОЙ МОДЕЛИ EFFICIENTNET.....	31
3.1. Проектирование программного продукта .....	31
3.1.1. Требования к проектируемому сервису .....	31

3.1.2. Проектирование схемы взаимодействия компонентов сервиса	31
3.2. Используемые технологии.....	33
3.2.1. Flask.....	33
3.2.2. API.....	33
3.2.3. HTML.....	34
3.2.4. CSS.....	34
3.2.5. Tensorflow.....	34
3.2.6. Keras.....	34
3.3. Программная реализация компонентов.....	35
3.3.1. Сохранение модели машинного обучения.....	35
3.3.2. Создание интерфейса для сервиса с использованием HTML и CSS	36
3.3.3. Создание REST API с помощью Flask.....	37
3.4. Пример работы разработанного сервиса.....	38

ГЛАВА 4. ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ ..... 41

4.1. Предпроектный анализ.....	41
4.1.1. Потенциальные потребители результатов исследования.....	41
4.1.2. Анализ конкурентных технических решений с позиции ресурсоэффективности и ресурсосбережения.....	42
4.1.3. SWOT-анализ.....	44
4.1.4. Оценка готовности проекта к коммерциализации.....	45
4.1.5. Методы коммерциализации результатов научно-технического исследования.....	47
4.2. Инициация проекта.....	48

4.2.1.	Цели и результат проекта.....	48
4.2.2.	Организационная структура проекта.....	50
4.2.3.	Ограничения и допущения проекта .....	51
4.3.	Планирование управления научно-техническим проектом .....	52
4.3.1.	Иерархическая структура работ проекта.....	52
4.3.2.	План проекта .....	52
4.4.	Бюджет научного исследования.....	56
4.4.1.	Сырье, материалы, покупные изделия и полуфабрикаты ....	56
4.4.2.	Затраты на электроэнергию .....	56
4.4.3.	Специальное оборудование для научных (экспериментальных) работ .....	56
4.4.4.	Основная заработная плата.....	57
4.4.5.	Дополнительная заработная плата.....	59
4.4.6.	Отчисления на социальные нужды .....	60
4.4.7.	Накладные расходы .....	61
4.4.8.	Формирование бюджетных расходов .....	62
4.5.	Оценка сравнительной эффективности исследования.....	62
4.6.	Выводы по разделу .....	65
	<b>ГЛАВА 5. СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ.....</b>	<b>66</b>
5.1.	Правовые и организационные вопросы обеспечения безопасности .....	66
5.1.1.	Специальные правовые нормы трудового законодательства	66
5.1.2.	Организационные мероприятия при компоновке рабочей зоны	67
5.2.	Производственная безопасность .....	68

5.2.1. Производственные факторы, связанные с недостаточной освещённостью рабочей зоны.....	70
5.2.2. Производственные факторы, связанные с аномальными микроклиматическими параметрами воздушной среды на местонахождении работающего .....	73
5.2.3. Производственные факторы, связанные с акустическими колебаниями в производственной среде и характеризующиеся неблагоприятными характеристиками шума .....	74
5.2.4. Производственные факторы, связанные с электромагнитными полями, неионизирующими ткани тела человека постоянного характера, связанного с повышенным образованием электростатических зарядов.....	74
5.2.5. Производственные факторы, связанные с перенапряжением анализаторов, вызванные информационной нагрузкой .....	75
5.2.6. Производственные факторы, связанные со статическими физическими перегрузками (нахождение в позе «сидя» или пребывание в вынужденной позе) .....	76
5.3. Экологическая безопасность .....	76
5.4. Безопасность в чрезвычайных ситуациях .....	77
5.5. Выводы по разделу .....	79
ЗАКЛЮЧЕНИЕ .....	80
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ .....	81
ПРИЛОЖЕНИЕ А Chapter 1. Analysis of the subject area.....	84
ПРИЛОЖЕНИЕ Б Обучение модели EfficientNetV4 на наборе больших данных .....	88

## ОСНОВНЫЕ СОКРАЩЕНИЯ И ОБОЗНАЧЕНИЯ

ПЭВМ — персональная электронно-вычислительная машина.

СНС — свёрточная нейронная сеть.

ANN — artificial neural network; искусственная нейронная сеть.

CNN — convolutional neural network; свёрточная нейронная сеть.

DL — deep learning; глубокое обучение.

ML — machine learning; машинное обучение.

SVM — support vector machine; машина опорных векторов.

SWOT — метод стратегического планирования, заключающийся в выявлении факторов внутренней и внешней среды организации и разделении их на четыре категории: Strengths (сильные стороны), Weaknesses (слабые стороны), Opportunities (возможности) и Threats (угрозы).

## ВВЕДЕНИЕ

В последние годы использование аналитики больших данных стало мощным инструментом для решения сложных проблем в различных областях, включая сельское хозяйство. Одной из ключевых задач, стоящих перед сельскохозяйственным сектором, является выявление болезней и насекомых-вредителей, которые могут нанести значительный ущерб посевам, и борьба с ними. Традиционные методы обнаружения этих угроз основаны на проверке вручную и могут отнимать много времени и быть подвержены ошибкам.

Объектом данного исследования являются подходы и методы распознавания заболеваний сельскохозяйственных культур. Предметом исследования являются модели нейронных сетей для распознавания заболеваний.

Целью данной работы является разработка сервиса, способного автоматически выявлять заболевания сельскохозяйственных культур на основе анализа больших данных.

Для достижения поставленной цели необходимо:

- Осуществить обзор существующих методов и алгоритмов обнаружения болезней сельскохозяйственных культур.
- Подготовить набор больших данных, связанных со здоровьем сельскохозяйственных культур.
- Реализовать алгоритм обнаружения болезней, который может эффективно обрабатывать большие данные.
- Обучить модель на больших данных и оценить ее производительность.
- Разработать на основе обученной модели сервис обнаружения болезней.



# ГЛАВА 1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

## 1.1. Описание предметной области

В связи со стремительным ростом населения планеты всё более острой становится тема обеспечения продуктами питания. Сельскохозяйственные культуры играют важнейшую роль в обеспечении продовольствием населения нашей планеты. На получение урожая от сельскохозяйственной деятельности человека влияет множество факторов, их можно разделить на три группы: объективные, субъективные и косвенные.

Субъективные факторы зависят от деятельности человека, и к ним можно отнести качество, вид и время основной и предпосевной обработки почвы; время, качество и способ посева; сорт, качество, выравненность и защищенность от вредителей и болезней семян; время, количество и качество внесения удобрений; количество, качество и время полива или осушения; своевременность и способы борьбы с сорными растениями, вредителями и болезнями; новые методы выращивания и агротехнические приемы и другие [1].

Существует несколько методов выявления болезней растений. Наиболее распространенные методы включают визуальную оценку болезней растений экспертами, микроскопическую оценку морфологических признаков для идентификации патогенов, а также молекулярные, серологические и микробиологические методы диагностики.

Однако ручное обнаружение болезней растений является трудоемким и подверженным ошибкам процессом. Оно может быть ненадежным методом выявления болезней растений и предотвращения их распространения. Передовые технологии, такие как машинное обучение (ML) и глубокое обучение (DL), могут преодолеть эти проблемы, обеспечивая раннее выявление болезней растений.

## **1.2. Обзор существующих решений**

Существует несколько причин, по которым наблюдается рост использования искусственного интеллекта для обнаружения болезней растений. Одна из них заключается в том, что решения на основе ИИ могут помочь в раннем обнаружении и предотвращении болезней сельскохозяйственных культур [2].

Многие современные модели/архитектуры DL эволюционировали после внедрения AlexNet – архитектуры сверточной нейронной сети для обнаружения изображений, их сегментации и классификации. В этом разделе представлены исследования, проведенные с использованием архитектур DL для идентификации и классификации болезней растений [3]. Описанные ниже алгоритмы используются для сравнения с разработанным алгоритмом.

### **1.2.1. ResNet-50 with SVM**

Mohameth и соавторы использовали различные современные архитектуры CNN и различные классификаторы для автоматического обнаружения болезней растений в наборе данных PlantVillage. Для извлечения признаков использовались архитектуры VGG-16, ResNet-50 и GoogLeNet CNN, в то время как для классификации использовались метод k-ближайших соседей и метод опорных векторов (SVM). В результате работы было установлено, что сочетание SVM с ResNet-50 показал наилучшие результаты с точностью 98% [4].

### **1.2.2. GoogLeNet**

Mohanty и соавторы проанализировали производительность архитектур CNN AlexNet и GoogLeNet при обнаружении болезней растений с использованием набора данных PlantVillage. Они провели 60 экспериментов с использованием 60 различных конфигураций. Результаты показали, что GoogLeNet с передаточным обучением показал наилучшие результаты с точностью 99,3%; согласно статье, количество параметров обучения, используемых GoogLeNet, составило около 27 миллионов [5].

### **1.2.3. Inception v3 transferred to target domain SVM**

Argüeso и соавторы применили алгоритм Few-shot learning (FSL) для классификации болезней растений. Они разделили изображения из набора данных PlantVillage на исходный и целевой домены. В исходном домене была использована тонкая настройка сети Inception V3, это позволило изучить общие характеристики листьев растений. Затем эти характеристики были перенесены в целевой домен для изучения новых типов листьев с использованием нескольких изображений. Для метода FSL была использована сиамская нейронная сеть с функцией Triplet loss. Результаты показали, что размер набора данных может быть уменьшен на 89,1% при потере точности всего на 4%, то есть этот метод хорош для небольших обучающих наборов [6].

### **1.3. Вывод по анализу предметной области**

Все рассмотренные выше исследовательские работы по распознаванию болезней растений достигли определенного уровня успеха в распознавании с высокой точностью. Однако, данные подходы к решению проблемы не лишены недостатков и ограничений:

1. Среди них широко используется набор данных PlantVillage – несмотря на то, что использование общего набора данных может облегчить сравнение различных моделей и методов, это также может привести к переобучению, если модели обучаются и оцениваются только на основе этого конкретного набора данных.
2. Сельскохозяйственные данные обширны и неструктурированы, поэтому для анализа наборов данных такого типа необходима среда с поддержкой больших данных.
3. Большинство рассмотренных исследований сосредоточено на конкретных видах заболеваний или конкретных видах растений.
4. Рассмотренные выше исследовательские работы имеют существенный недостаток, заключающийся в том, что во всех исследовательских работах использовалось очень большое количество параметров обучения.

В связи с этим, огромное значение имеют дополнительные исследования для разработки надежных моделей, которые будут работать для различных видов растений и болезней.

## **ГЛАВА 2. РАЗРАБОТКА МЕТОДА ОБНАРУЖЕНИЯ БОЛЕЗНЕЙ РАСТЕНИЙ**

### **2.1. Описание используемой архитектуры нейронной сети**

Сверточные нейронные сети (CNN) обычно разрабатываются с учетом ограниченных ресурсов, а затем масштабируются для улучшения точности при наличии большего количества ресурсов. Например, ResNet можно масштабировать с ResNet-18 до ResNet-200, увеличив количество слоев. Часто используемые методы масштабирования включают увеличение глубины или ширины CNN или использование большего разрешения входных изображений для обучения и оценки. Однако эти методы могут потребовать ручной настройки и не всегда обеспечивают оптимальную производительность.

В статье “Efficientnet: Rethinking model scaling for convolutional neural networks” был представлен новый метод масштабирования моделей сверточных нейронных сетей с использованием составного коэффициента для структурированного масштабирования. В отличие от традиционных методов, которые масштабируют размеры сети произвольно, модели семейства EfficientNet достигают более эффективных результатов за счет равномерного масштабирования глубины, ширины и разрешения при одновременном уменьшении масштаба модели [7].

В данном разделе рассматриваются ключевые характеристики модели EfficientNet и то, как она достигает высокой производительности. Описаны ее архитектура, ключевые компоненты и методы масштабирования, используемые для повышения точности и эффективности. Кроме того, рассматриваются результаты, показанные моделью на различных наборах данных, и проводится сравнение с другими распространенными архитектурами сверточных нейронных сетей.

#### **2.1.1. Метод составного масштабирования**

При разработке метода составного масштабирования авторы провели систематическое исследование влияния, которое каждый метод

масштабирования оказывает на производительность модели. Они пришли к выводу, что, хотя масштабирование отдельных измерений помогает повысить производительность модели, наилучший результат достигается при балансировании масштаба во всех трех измерениях – ширине, глубине и разрешении изображения.

Первым шагом в методе комплексного масштабирования является выполнение поиска по сетке, для нахождения взаимосвязей между различными масштабируемыми параметрами базовой сети при фиксированном ограничении ресурсов. Это позволяет определить коэффициенты масштабирования для каждого измерения. Затем эти коэффициенты используются для масштабирования базовой сети до желаемого размера [7].

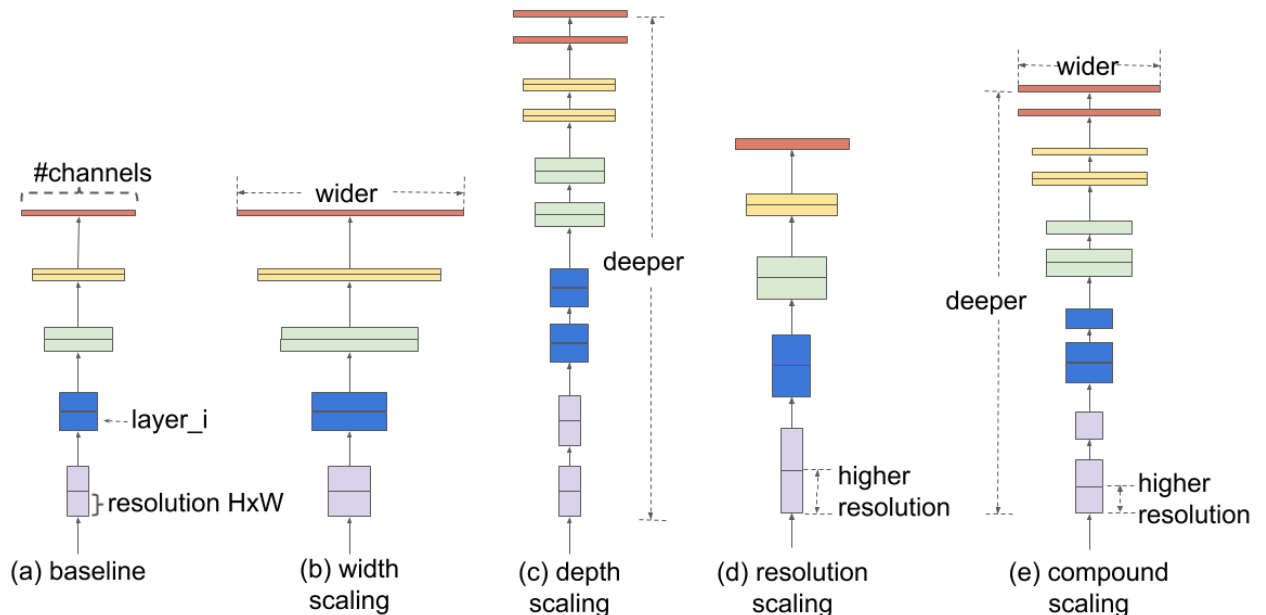


Рисунок 1 – Сравнение методов масштабирования (b-d, где d — предложенный метод), а — начальная архитектура

### 2.1.2. Архитектура модели EfficientNet

Эффективность масштабирования нейронной сети зависит от ее начальной архитектуры. Для повышения производительности нейронной сети исследователи использовали фреймворк AutoML – MNAS, для автоматического выбора начальной архитектуры, где при выборе устройства нейронной сети оптимизировались как точность модели, так и размер модели (FLOPS).

Архитектура использует мобильную обращенную свертку с узким местом – MBConv, схожую с MobileNetV2 и MnasNet. Эта начальная модель затем масштабировалась и породила класс EfficientNet-моделей [7].

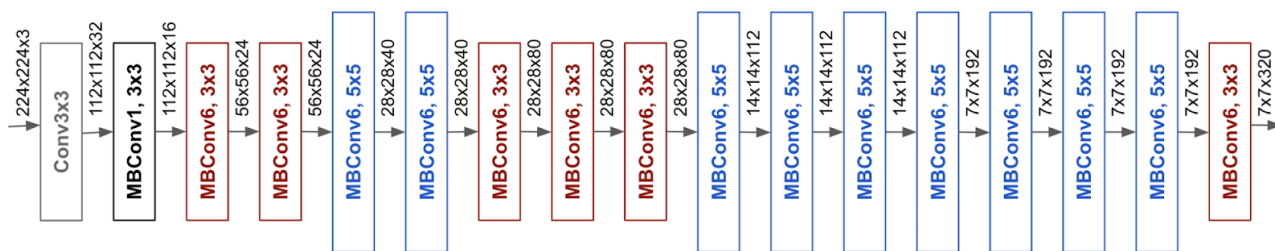


Рисунок 2 – Архитектура начальной модели EfficientNet-B0

### 2.1.3. Производительность модели EfficientNet

Рисунок 3 показывает производительность EfficientNet по сравнению с другими архитектурами сетей. Самая большая модель EfficientNet B7 показала лучшие результаты на наборах данных ImageNet и CIFAR100. Модель достигла примерно 84,4% точности top-1, что означает, что наиболее вероятный ответ модели совпадает с правильным, и 97,3% точности top-5, что означает, что правильный ответ входит в пять наиболее вероятных ответов. Кроме того, размер модели EfficientNet был в 8,4 раза меньше, а скорость работы в 6,1 раза выше, чем у предыдущей лучшей модели CNN. Она достигла 91,7% точности на наборе данных CIFAR-100 и 98,8% точности на наборе данных Flowers [7].

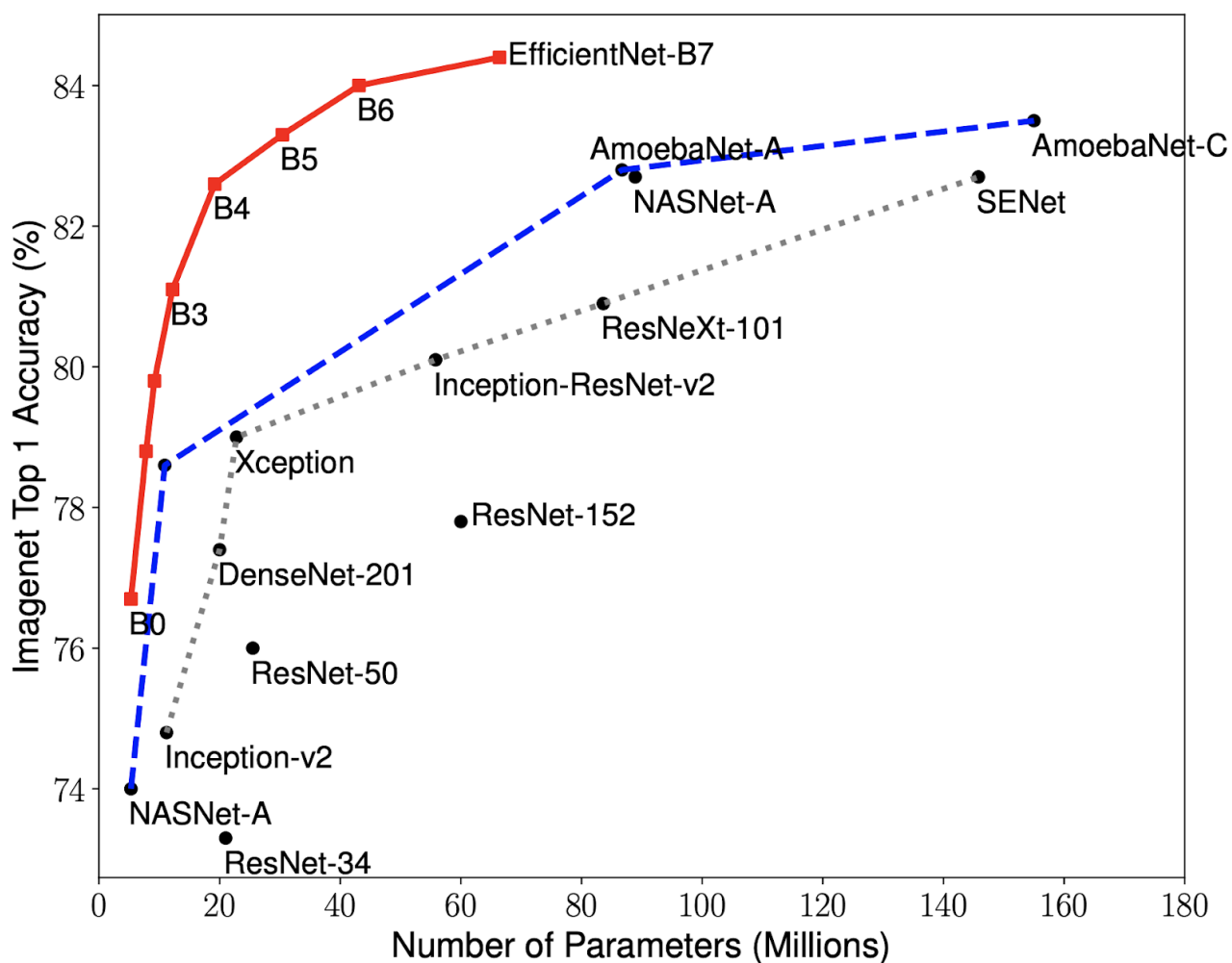


Рисунок 3 – Сравнение точности и размера сверточных архитектур и EfficientNets

В целом, модели EfficientNet обеспечивают как более высокую точность, так и более высокую эффективность по сравнению с существующими CNN, уменьшая размер параметров и сбой.

## 2.2.Используемый набор данных

Как упоминалось выше, большая часть исследований алгоритмов обнаружения болезней сельскохозяйственных культур основана на наборе данных PlantVillage. Одно из главных критических замечаний к набору данных Plant Village заключается в том, что он содержит шум, который коррелирует с метками, модели глубокого обучения могут легко использовать это смещение для составления прогнозов [8]. Смещение набора данных – это систематические ошибки в наборе данных, которые могут повлиять на точность и объективность



модели машинного обучения, обученной на этих данных. Ошибки могут возникать по ряду причин, включая ошибки выборки, ошибки измерения и ошибки маркировки.

Использование большого набора данных может уменьшить смещение данных, предоставляя больший и разнообразный набор данных для обучения моделей машинного обучения.

Большие данные – это разнообразные данные, поступающие с высокой скоростью, объем которых постоянно растет. Эти три свойства являются основными для больших данных.

Количество данных – важный фактор. При больших объемах приходится иметь дело с большим количеством неструктурированных данных низкой плотности.

Скорость в данном контексте – это скорость приема данных и, возможно, действий на их основе. Большие данные обновляются регулярно, поэтому необходимы интеллектуальные технологии для их обработки.

Разнообразие означает, что доступные данные принадлежат к разным типам. Данные могут быть структурированными, полуструктурированными или неструктурированными.

В качестве набора больших данных в данной работе используется PlantDisease Dataset [9]. Он содержит 50 классов с более чем 76 000 изображений общим размером 17,4 ГБ.



Рисунок 4 – Примеры изображений в обучающем наборе

### 2.3. Обучение модели

Набор данных, использованный в этом исследовании, был разделен на два отдельных набора: обучающий и тестирующий в соотношении 80% к 20% соответственно. Обучающий набор был использован для обучения модели, тестирующий – для проверки ее работоспособности. Эксперимент проводился в течение 20 итераций в 40 партиях. Модель EfficientNetB4 инициализируется с предварительно обученными весами ImageNet и максимальным объединением (пулингом). Используется значение параметра нормализация равное 0,99 и темп обучения 0,001. Плотные слои инициализируются с размерностью внешнего пространства, равной 256. Используются функции регуляризации L2 с коэффициентом 0,016 и L1 с коэффициентом 0,006. Для этих слоев используется функция активации ReLU.

## 2.4. Оценка показателей производительности разработанной модели

Производительность модели измерялась с использованием показателей меткости (Accuracy) – выражение (1), точности (Precision) – выражение (2), полноты (Recall) – выражение (3) и F1-меры (F1-score) – выражение (4).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \quad (1)$$

Где  $TP$  — истинно-положительные значения;

$TN$  — истинно-отрицательные значения.

$FP$  — ложно-положительные значения.

$FN$  — ложно-отрицательные значения.

$$Precision = \frac{TP}{TP + FP}, \quad (2)$$

Где  $TP$  — истинно-положительные значения.

$FP$  — ложно-положительные значения.

$$Recall = \frac{TP}{TP + FN}, \quad (3)$$

Где  $TP$  — истинно-положительные значения.

$FN$  — ложно-отрицательные значения.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}, \quad (4)$$

Таблица 1 – Оценка показателей производительности

Class	TP	TN	FP	FN	Acc (%)	Pre (%)	Recall (%)	F-1
Apple scab	50	1900	0	0	100.00	100.00	100.00	100.00
Apple Black rot	50	1900	0	0	100.00	100.00	100.00	100.00
Apple Cedar apple rust	50	1900	0	0	100.00	100.00	100.00	100.00
Apple healthy	50	1900	0	0	100.00	100.00	100.00	100.00
Background without leaves	50	1898	2	0	99.90	96.15	100.00	98.03
Blueberry healthy	50	1900	0	0	100.00	100.00	100.00	100.00

Cherry Powdery mildew	50	1900	0	0	100.00	100.00	100.00	100.00
Cherry healthy	49	1900	0	1	99.95	100.00	98.00	98.98
Corn Cercospora leaf spot Gray leaf spot	49	1898	2	1	99.85	96.08	98.00	97.03
Corn Common rust	50	1900	0	0	100.00	100.00	100.00	100.00
Corn Northern Leaf Blight	48	1899	1	2	99.85	97.96	96.00	96.97
Corn healthy	50	1900	0	0	100.00	100.00	100.00	100.00
Grape Black rot	50	1899	1	0	99.95	98.04	100.00	99.01
Grape Esca (Black Measles)	49	1900	0	1	99.95	100.00	98.00	98.98
Grape Leaf blight (Isariopsis Leaf Spot)	50	1900	0	0	100.00	100.00	100.00	100.00
Grape healthy	50	1900	0		100.00	100.00	100.00	100.00
Orange Haunglongbing (Citrus greening)	50	1900	0	0	100.00	100.00	100.00	100.00
Peach Bacterial spot	50	1900	0	0	100.00	100.00	100.00	100.00
Peach healthy	50	1900	0	0	100.00	100.00	100.00	100.00
Pepper, bell Bacterial spot	50	1900	0	0	100.00	100.00	100.00	100.00
Pepper, bell healthy	50	1900	0	0	100.00	100.00	100.00	100.00
Potato Early blight	50	1900	0	0	100.00	100.00	100.00	100.00
Potato Late blight	50	1899	1	0	99.95	98.04	100.00	99.01
Potato healthy	50	1899	1	0	99.95	98.04	100.00	99.01
Raspberry healthy	50	1900	0	0	100.00	100.00	100.00	100.00
Soybean healthy	50	1900	0	0	100.00	100.00	100.00	100.00
Squash Powdery mildew	50	1900	0	0	100.00	100.00	100.00	100.00
Strawberry Leaf scorch	50	1900	0	0	100.00	100.00	100.00	100.00
Strawberry healthy	50	1900	0	0	100.00	100.00	100.00	100.00
Tomato Bacterial spot	48	1900	0	2	99.90	100.00	96.00	97.95
Tomato Early blight	49	1899	1	1	99.90	98.00	98.00	98.00
Tomato Late blight	49	1899	1	1	99.90	98.00	98.00	98.00
Tomato Leaf Mold	48	1900	0	2	99.90	100.00	96.00	97.95
Tomato Septoria leaf spot	50	1900	0	0	100.00	100.00	100.00	100.00
Tomato Spider mites Two-spotted spider	50	1899	1	0	99.95	98.04	100.00	99.01

mite								
Tomato Target Spot	49	1900	0	1	99.95	100.00	98.00	98.98
Tomato Yellow Leaf Curl Virus	50	1899	1	0	99.95	98.04	100.00	99.01
Tomato mosaic virus	50	1900	0	0	100.00	100.00	100.00	100.00
Tomato healthy	50	1900	0	0	100.00	100.00	100.00	100.00

Тестовый набор был использован для проверки работоспособности модели EfficientNet с использованием точности обучения и валидации и потерь. Категориальная перекрестная энтропия использовалась в качестве функции потерь.

На рисунке 5 показан график точности обучения и валидации. Обе кривые графика приближаются друг к другу по мере продвижения эпох. Эпоха представляет собой одну итерацию обучения модели со всеми обучающими данными.

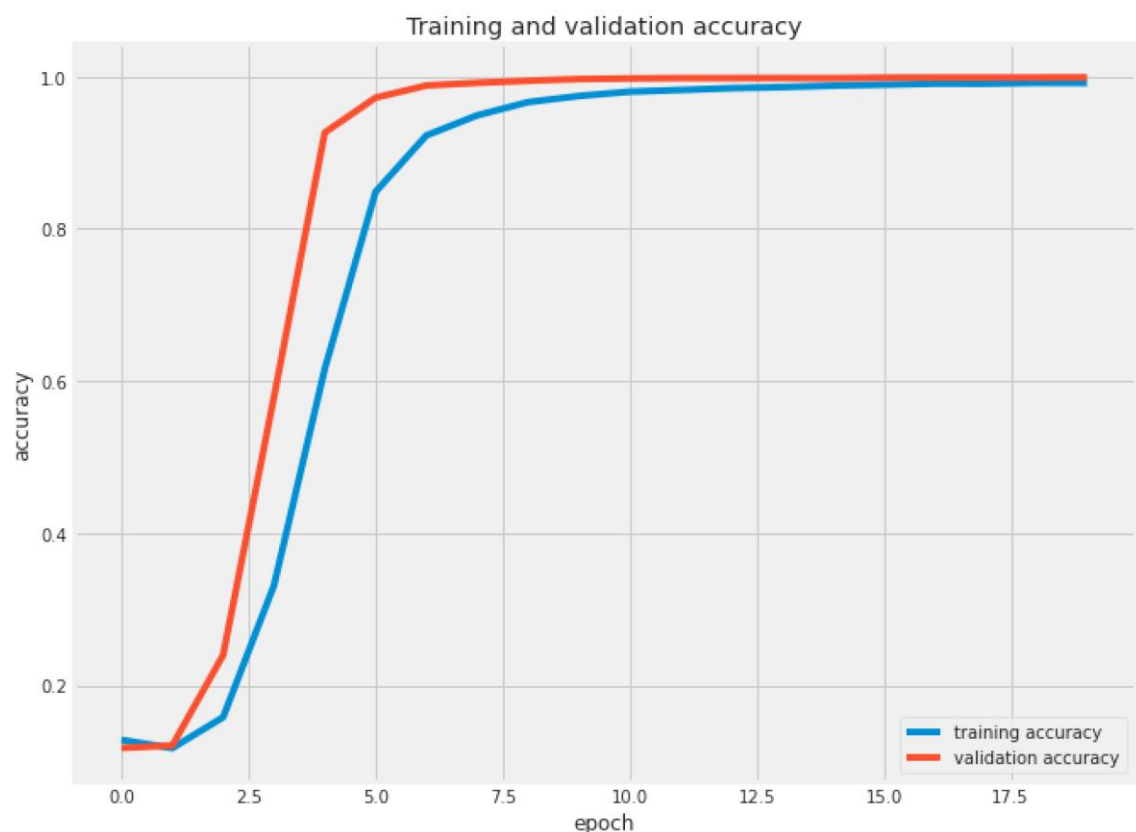


Рисунок 5 – Точность обучения и валидации

Аналогично, потери при обучении и валидации, показанные на рисунке 6, демонстрируют обоснованность предложенной модели в том смысле, что две кривые приближаются друг к другу, прогрессируют одновременно по мере прохождения эпох, совпадают на эпохе 13 и прогрессируют одинаково. Это указывает на отсутствие переобучения модели EfficientNet.

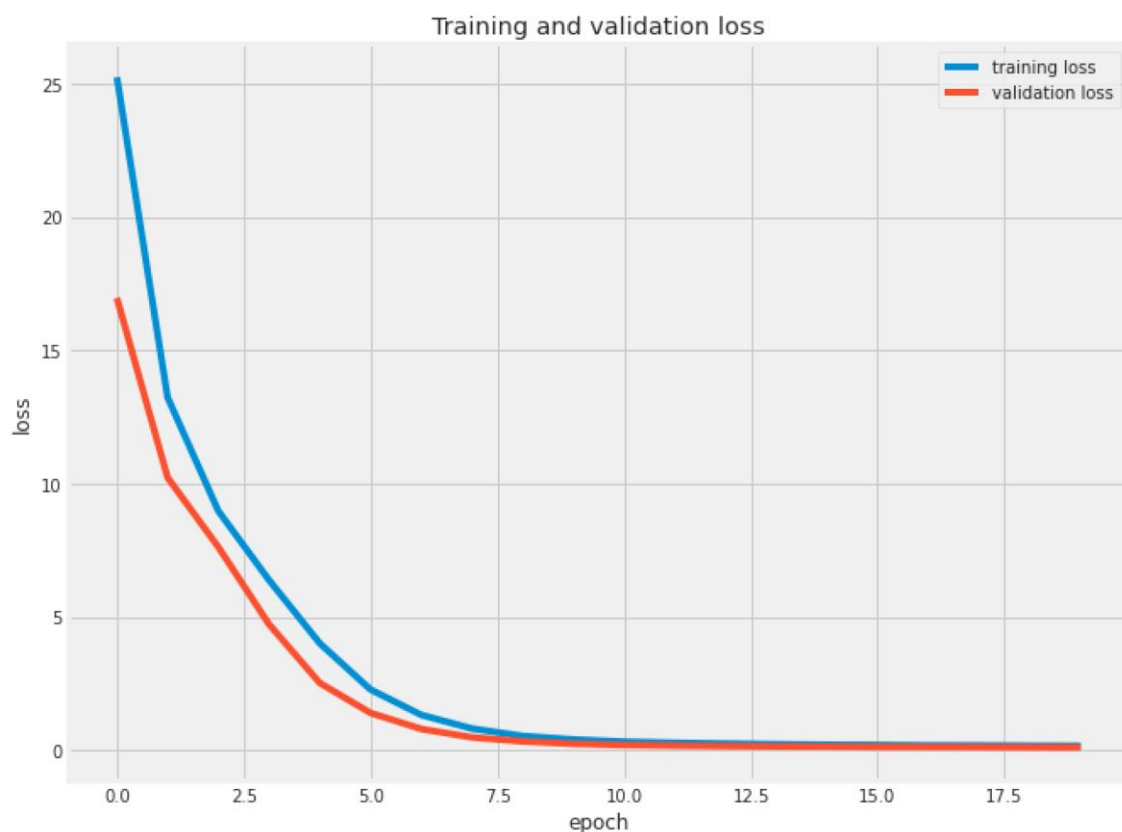


Рисунок 6 – Потери обучения и валидации

## **ГЛАВА 3. РАЗРАБОТКА СЕРВИСА НА ОСНОВЕ ОБУЧЕННОЙ МОДЕЛИ EFFICIENTNET**

### **3.1. Проектирование программного продукта**

#### **3.1.1. Требования к проектируемому сервису**

Тщательно сформулированные и задокументированные требования к сервису необходимы для правильного понимания его концепции и возможностей. Для разрабатываемого сервиса были определены следующие общие требования:

1. Проект должен быть разработан на языке Python.
2. В качестве фреймворка для написания REST API использовать Flask.
3. Сервис должен иметь высокую точность распознавания болезней растений на изображениях.
4. Сервис должен быть простым в использовании для пользователей без специальных знаний в области сельского хозяйства или информационных технологий.
5. Сервис должен быстро обрабатывать изображения и предоставлять результаты в короткие сроки.
6. Сервис должен обеспечивать безопасность и конфиденциальность пользовательских данных, включая загружаемые изображения и полученные результаты.
7. Сервис должен поддерживать загрузку изображений в различных форматах, таких как JPEG, PNG и других.

#### **3.1.2. Проектирование схемы взаимодействия компонентов сервиса**

Данная часть работы посвящена проектированию схем взаимодействия сервисных компонентов с использованием Flask и моделей искусственного интеллекта. Flask – это микрофреймворк для создания веб-приложений с использованием языка Python, который позволяет легко обрабатывать HTTP-запросы.

Модели искусственного интеллекта могут быть использованы для обработки данных и принятия решений внутри сервиса. В данном разделе рассмотрены основные принципы и методы проектирования схемы взаимодействия компонентов, а также способы интеграции этих компонентов для создания эффективного и надежного сервиса.

Этот проект состоит из трех частей:

1. model.h5 – содержит код обученной модели нейронной сети архитектуры EfficientNet, который позволяет определять наличие заболеваний растений на изображениях.
2. app.py – содержит API-интерфейс Flask, который получает изображения растений через GUI, вычисляет предсказанное значение на основе обученной модели и возвращают его.
3. HTML/CSS – содержит HTML-шаблон и стили CSS для того, чтобы пользователь мог загружать изображения растений для их дальнейшего анализа.

Общая схема взаимодействия всех частей сервиса представлена на рисунке 7.

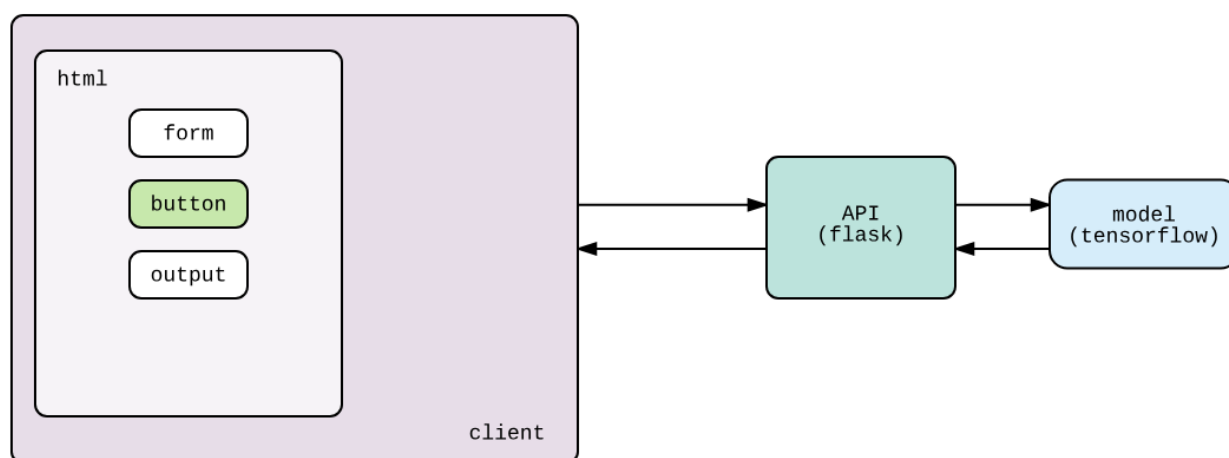


Рисунок 7 – Схема взаимодействия компонентов сервиса



## **3.2.Используемые технологии**

### **3.2.1. Flask**

Flask – это микрофреймворк для создания простых и быстрых проектов на языке Python с возможностью масштабирования до более сложных приложений. Термин “микрофреймворк” означает, что он не содержит встроенных инструментов и библиотек, которые разработчик может установить самостоятельно в зависимости от требований проекта [10].

Flask поддерживает различные протоколы HTTP-запросов и имеет встроенный сервер разработки, который облегчает тестирование и отладку приложений. Он также легко интегрируется со сторонними библиотеками и расширениями, предоставляя разработчикам доступ к обширной экосистеме инструментов и ресурсов, которые они могут использовать для создания надежных и масштабируемых приложений.

Популярность Flask, широко используемого для разработки RESTful API, микросервисов и веб-приложений, обусловлена его простотой и удобством использования. Это позволяет разработчикам сосредоточиться на создании основных функциональных возможностей своих приложений, не беспокоясь о базовой инфраструктуре. Это делает его отличным выбором для проектов с жесткими сроками и ограниченными ресурсами.

### **3.2.2. API**

API (Application Programming Interface) – набор определений и функций, который позволяет приложениям получать доступ к данным и взаимодействовать с внешними программными компонентами, операционными системами или микросервисами.

Существуют несколько видов API. В данном проекте реализуется работа с REST API. REST API взаимодействует при помощи HTTP запросов, выполняя стандартные функции: создание, обновление, чтение, удаление записей в ресурсе.

### **3.2.3. HTML**

HTML (HyperText Markup Language) – стандартный язык разметки для создания веб-страниц. Он позволяет создавать и структурировать разделы, абзацы и ссылки с помощью HTML-элементов (строительных блоков веб-страницы), таких как теги и атрибуты. HTML-элементы – строительные блоки веб-страницы. Тег сообщает веб-браузеру, где начинается и заканчивается элемент, в то время как атрибут описывает характеристики элемента [11].

### **3.2.4. CSS**

CSS (Cascading Style Sheets) расшифровывается как каскадные таблицы стилей, данный язык используется для стилизации элементов, написанных на языке разметки, таком как HTML [12].

CSS позволяет определять шрифт, цвет, размеры и расположение блоков на веб-странице. Основная цель CSS – разделить описание логической структуры веб-страницы (с помощью HTML или других языков разметки) от описания ее внешнего вида. Это разделение делает разработку более структурированной и удобной для восприятия.

### **3.2.5. Tensorflow**

TensorFlow – это открытая библиотека машинного обучения, созданная Google для решения задач по созданию и обучению нейронных сетей с целью автоматического распознавания и классификации образов, достигая качества человеческого восприятия [13]. Основной API реализован для языка Python.

TensorFlow позволяет пользователям отвлекаться от технических деталей машинного обучения и сосредоточиться на непосредственном решении своих задач. Это мощный инструмент, который позволяет создавать, обучать и использовать нейронные сети различных типов.

### **3.2.6. Keras**

Keras – это открытая нейросетевая библиотека на языке Python, которая обеспечивает простое создание нейронных сетей [14]. Она совместима с такими платформами как TensorFlow, Microsoft Cognitive Toolkit, Theano и MXNet.

TensorFlow и Theano являются наиболее часто используемыми численными платформами для разработки алгоритмов глубокого обучения на Python, но они могут быть сложными в использовании. Keras направлена на быструю работу с глубокими нейронными сетями и разработана таким образом, чтобы быть компактной, модульной и расширяемой.

На основе вышеизложенного было принято решение использовать популярную комбинацию Keras и TensorFlow для обучения и работы со сверточными нейронными сетями.

### **3.3. Программная реализация компонентов**

#### **3.3.1. Сохранение модели машинного обучения**

После того, как модель машинного обучения была обучена, полезно сохранить модель со всеми ее параметрами. После сохранения обученная модель может быть загружена по мере необходимости, и классификация может быть выполнена без повторного обучения модели с нуля. Это экономит как время, так и системные ресурсы, что приводит к более быстрой классификации. Модель машинного обучения Keras состоит из следующих компонентов:

1. Архитектура, или конфигурация (определяет, какие слои содержит модель и как они связаны друг с другом).
2. Набор весовых значений.
3. Оптимизатор (определяется во время компиляции модели)
4. Набор потерь и метрик (определяется во время компиляции модели или вызовом функций `add_loss()` или `add_metric()`).

API Keras позволяет сохранять все эти фрагменты на диск сразу или выборочно сохранять только некоторые из них: сохранять все в один архив в формате модели TensorFlow (или в более старом формате Keras – H5) и сохранять только архитектуру / конфигурацию в виде файла JSON.

### 3.3.2. Создание интерфейса для сервиса с использованием HTML и CSS

Веб-формы – это одна из основных точек взаимодействия между пользователем и веб-сайтом или приложением. Формы позволяют пользователям вводить данные, которые обычно отправляются на веб-сервер для обработки и хранения или используются на стороне клиента для немедленного обновления интерфейса каким-либо образом.

HTML веб-формы состоит из одного или нескольких элементов управления формой (иногда называемых виджетами), а также некоторых дополнительных элементов, помогающих структурировать форму в целом - их часто называют HTML-формами. Элементы управления могут быть однострочными или многострочными текстовыми полями, выпадающими списками, кнопками, флажками или переключателями и в основном создаются с использованием элемента `<input>`. Элементы управления формы также могут быть запрограммированы для обеспечения ввода определенных форматов или значений.

Предварительная схема основных элементов веб-страницы представлена на рисунке 8.

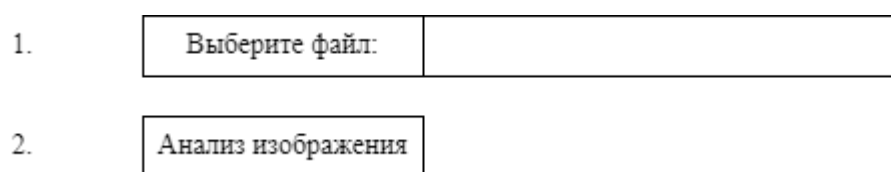


Рисунок 8 – Основные элементы веб-страницы

В данном случае, элемент 1 определяет поле выбора файла и кнопку "Обзор" для загрузки файлов, реализуется с помощью HTML-элемента `<input type="file">`.

Элемент 2 представляет собой кнопку, по нажатию на которую, загруженное пользователем изображение отправляется на анализ. Для реализации данной кнопки используется HTML-элемент `<input type="submit">`.

### 3.3.3. Создание REST API с помощью Flask

Для реализации серверной части сервиса использовалась библиотека Flask для создания REST API, который обеспечивает прямой доступ к серверным методам, независимо от реализованного пользовательского интерфейса. Таким образом, один и тот же API можно вызывать и использовать с разными интерфейсными платформами.

Пользователи могут использовать интерфейс для загрузки выбранных изображений через веб-страницу, выполнив запрос POST API. Когда запрос POST отправляется на URL-адрес с изображением в качестве содержимого, сервер выполняет следующие действия:

Пользователь может загрузить изображение через веб-страницу, используя интерфейс, выполнив POST-запрос API. При отправке POST-запроса на URL-адрес с изображением в качестве содержимого сервер выполняет следующие действия.

1. Сохранение загруженного изображения во временное хранилище на сервере. Затем это местоположение индексируется во временную переменную, которая сбрасывается после завершения классификации.
2. Сервер загружает модель машинного обучения, которая была предварительно обучена.
3. Размер загруженного изображения изменяется в соответствии с входными данными, ожидаемыми моделью.
4. Модель классифицирует изображения и указывает прогноз заболевания. Результаты состоят из массива вероятностей. Затем находится метка класса с наибольшей вероятностью и сопоставляется с соответствующей классификацией.
5. Результаты отправляются во внешний интерфейс в виде ответа JSON; объект JSON содержит класс с наибольшей вероятностью, который представляет собой окончательный прогноз.

### 3.4. Пример работы разработанного сервиса

Окончательный вариант разработанного интерфейса сервиса представлен на рисунке 9, он состоит из следующих элементов:

1. Заголовок веб-страницы.
2. Краткое описание возможностей по взаимодействию с сервисом.
3. Поле выбора файла и кнопку "Обзор" для загрузки файлов.
4. Кнопка, при нажатии на которую изображение, загруженное пользователем, отправляется на анализ.

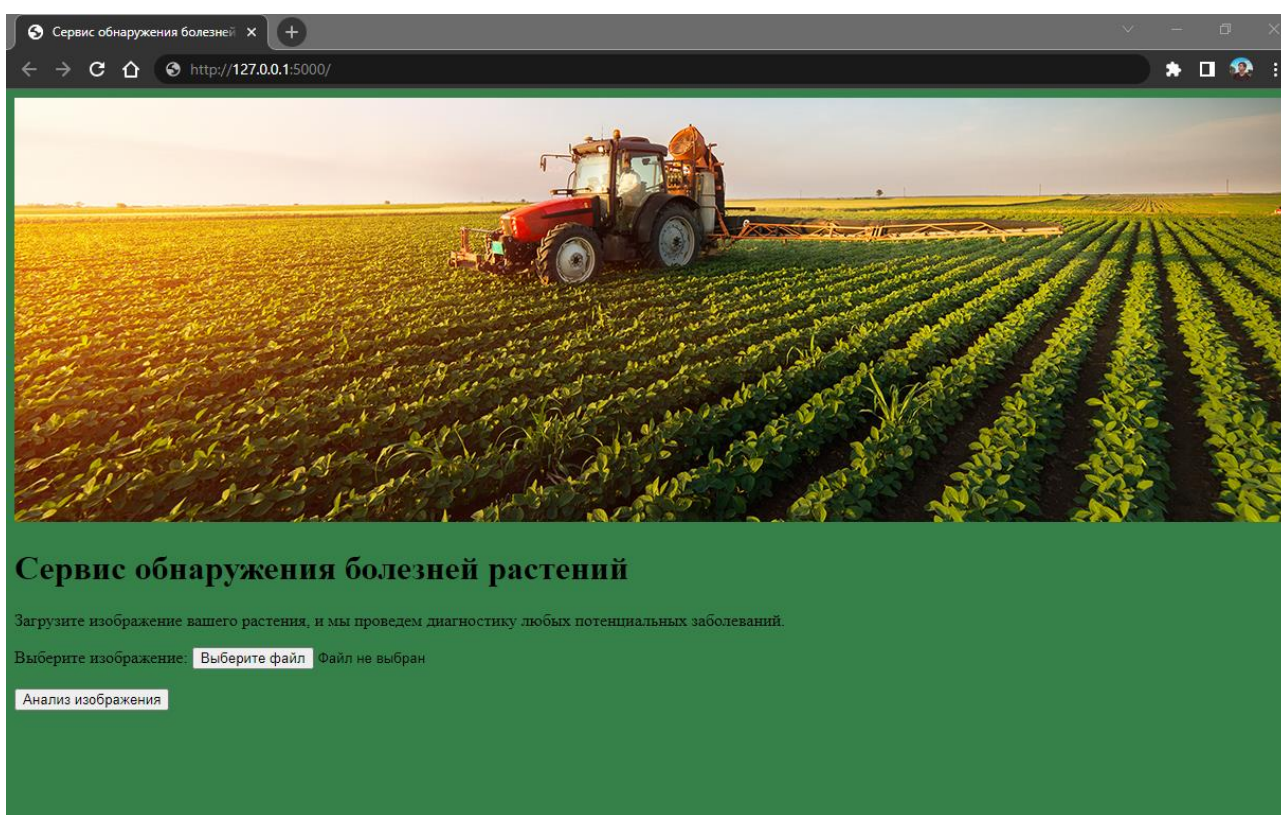


Рисунок 9 – Основное окно сервиса

При нажатии на кнопку “Выберите файл” открывается окно проводника операционной системы, в котором пользователь выбирает загружаемый им файл.

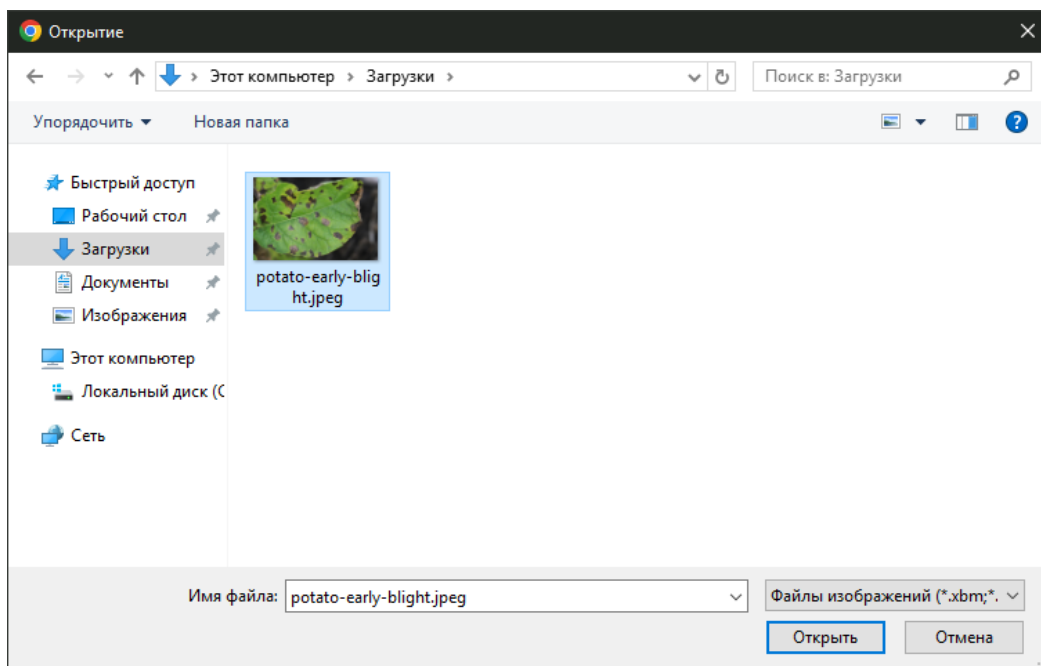


Рисунок 10 – Выбор изображения для анализа

Выбранное пользователем изображение для анализа также отобразится на странице веб-сервиса.

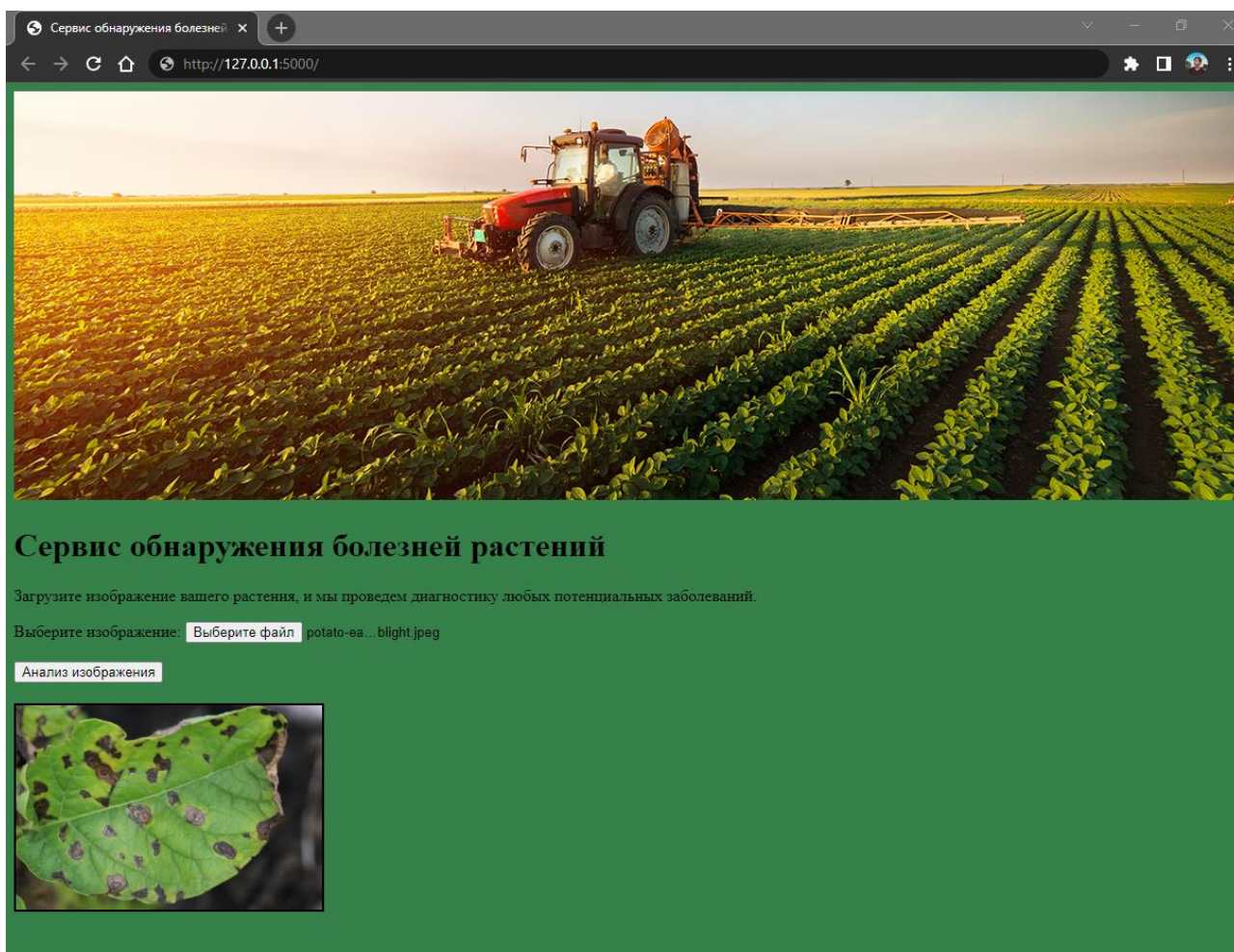
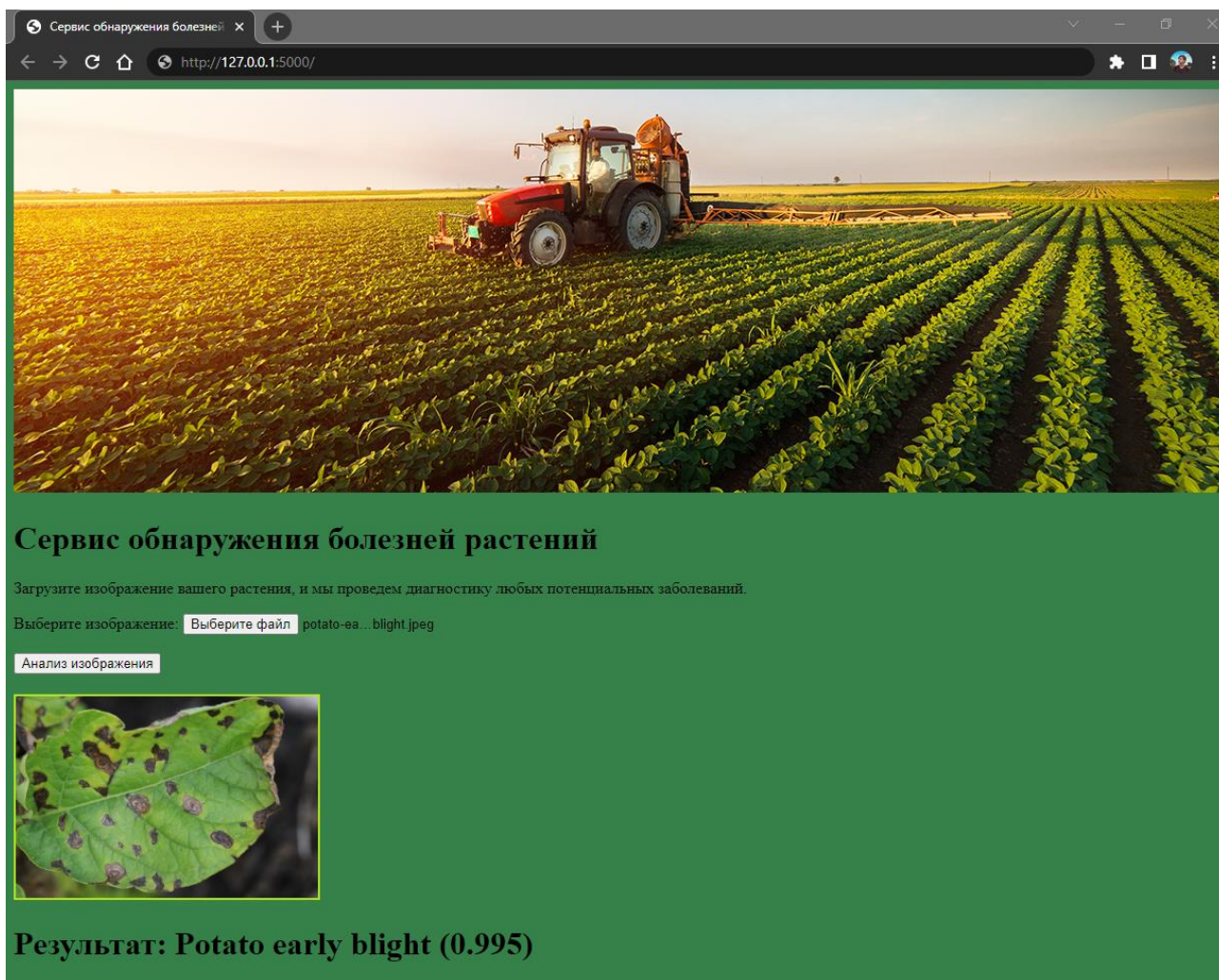


Рисунок 11 – Загрузка изображения для анализа

По нажатию на кнопку “Анализ изображения” с помощью API на основе Flask отправляется запрос на анализ изображения с помощью заранее сохраненной модели нейронной сети.


Результат определения наличия заболевания, включающий в себя класс заболевания, а также вероятную точность определения, отображаются на странице веб-сервиса.



Сервис обнаружения болезней растений

Загрузите изображение вашего растения, и мы проведем диагностику любых потенциальных заболеваний.

Выберите изображение:



**Результат: Potato early blight (0.995)**

Рисунок 12 – Результат анализа изображения



## **ГЛАВА 4. ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ**

### **4.1. Предпроектный анализ**

#### **4.1.1. Потенциальные потребители результатов исследования**

Целью данной работы является создание сервиса, способного автоматически выявлять заболевания сельскохозяйственных культур на изображениях.

Разработка сервиса обнаружения болезней сельскохозяйственных культур представляет собой создание программного продукта, который позволит определять наличие заболеваний у растений на основе анализа изображений с использованием архитектуры нейронных сетей EfficientNet. Для обучения моделей СНС используется набор данных, содержащий изображения здоровых и больных растений сельскохозяйственных культур. Результатом вычисления модели нейронной сети для каждого изображения является определение наличия или отсутствия заболевания и его типа.

Разработанный сервис может быть использован на предприятиях сельского хозяйства, занимающихся полеводством, овощеводством и садоводством, для раннего выявления заболеваний.

Для того чтобы провести анализ потребителей результатов исследования необходимо рассмотреть целевой рынок и провести его сегментирование. Целевой рынок представляет собой сегмент рынка, на котором разработанный программный продукт может быть продан в будущем. Сегментирование – это разделение покупателей на однородные группы, каждая из которых может нуждаться в определенном продукте.

Так как главными потребителями разработки являются сельскохозяйственные предприятия, в качестве основных критериев сегментации были выделены отрасль деятельности компании и её размер. Результат сегментации представлен в таблице 2.

Таблица 2 – Карта сегментирования рынка

		Тип деятельности		
		Полеводство	Овощеводство	Садоводство
Размер предприятия	Большой			
	Средний			
	Малый			

АО Агрофирма "Черноморец"	ООО Садовая группа "Велес"	ООО "Обрий"
---------------------------	----------------------------	-------------

Из карты сегментирования рынка услуг по использованию алгоритмов обнаружения заболеваний сельскохозяйственных культур видно, что основными потребителями являются малые и средние предприятия в сфере полеводства, овощеводства и садоводства.

#### **4.1.2. Анализ конкурентных технических решений с позиции ресурсоэффективности и ресурсосбережения**

В качестве основных конкурентных технических решений были выбраны следующие программные продукты:

- проект, предлагаемый в текущем исследовании (в таблице обозначен как «1»);
- проект, который использует архитектуру нейронной сети GoogLeNet. (в таблице обозначен как «2»);
- проект, который использует архитектуру нейронной сети AlexNet. (в таблице обозначен как «3»).

Экспертная оценка основных технических характеристик этих решений представлена в таблице 3. В качестве набора технических критериев рассматриваются следующие требования:

1. Точность — точность обнаружения (классификации) объектов;
2. Качество распознавания — степень пригодности архитектуры к решению задачи;
3. Универсальность — насколько универсальна архитектура;
4. Востребованность — насколько востребована архитектура;
5. Требовательность к данным — насколько требовательна обученная модель к данным (изображениям);
6. Доступность — насколько открытой является разработка;
7. Цена — насколько дорогой является разработка.

Таблица 3 – Результаты конкурентного анализа

Критерии оценки	Вес критерия, $V_i$	Баллы			Конкурентоспособность		
		Б <sub>1</sub>	Б <sub>2</sub>	Б <sub>3</sub>	К <sub>1</sub>	К <sub>2</sub>	К <sub>3</sub>
<b>Технические критерии оценки ресурсоэффективности</b>							
1. Точность	0,25	5	3	4	1,25	0,75	1
2. Качество распознавания	0,2	4	3	3	0,8	0,6	0,6
3. Универсальность	0,1	5	1	2	0,5	0,1	0,2
4. Востребованность	0,1	4	2	3	0,4	0,2	0,3
5. Требовательность к данным	0,15	5	2	3	0,75	0,3	0,45
<b>Экономические критерии оценки эффективности</b>							
1. Доступность	0,1	4	3	4	0,4	0,3	0,4
2. Цена	0,1	4	3	3	0,4	0,3	0,3
<b>Итого</b>	1	31	17	22	4,5	2,55	3,25

По результатам проведённого конкурентного анализа можно сделать вывод, что предложенный метод наиболее конкурентоспособен в сравнении с другими рассмотренными методами. Такая конкурентоспособность обусловлена за счёт универсальности метода, а также за счёт обеспечения высокой точности обнаружения болезней.

### 4.1.3. SWOT-анализ

SWOT – Strengths (сильные стороны), Weaknesses (слабые стороны), Opportunities (возможности) и Threats (угрозы) – это комплексный анализ научно-исследовательского проекта. Этот анализ используется для изучения внешней и внутренней среды проекта. Первый этап заключается в описании сильных и слабых сторон проекта и определяет возможности и угрозы, которые возникли или могут возникнуть в проекте. Второй этап выражается в определении совместимости сильных и слабых сторон проекта с внешней средой, а также то, какие стратегические изменения необходимо провести.

В результате SWOT-анализа была составлена итоговая матрица (таблица 4), описывающая сильные и слабые стороны проекта, выявленные возможности и угрозы, а также их корреляцию.

Таблица 4 – Результаты SWOT-анализа

	<p>Сильные стороны:</p> <p>С1. Высокая точность обнаружения</p> <p>С2. Компактность используемой нейросетевой архитектуры</p> <p>С3. Возможность дополнительного обучения модели.</p>	<p>Слабые стороны:</p> <p>Сл1. Необходимость наличия большой базы данных для обучения модели</p> <p>Сл2. Отсутствие финансирования.</p> <p>Сл3. Потребность в вычислительных ресурсах при переобучении.</p>
<p>Возможности:</p> <p>В1. Тенденция к более масштабному использованию нейросетевых алгоритмов для решения задач обнаружения</p> <p>В2. Возможность улучшения и масштабирования проекта</p>	<p>В1С1:</p> <p>В последнее время растет интерес к алгоритмам на основе нейронных сетей, появляются новые архитектуры нейронных сетей, которые в будущем можно будет легко интегрировать в имеющуюся разработку, тем самым значительно улучшив её конкурентоспособность.</p>	<p>Вл1Сл1:</p> <p>Для увеличения точности обнаружения необходимо иметь большую базу данных изображений, а также затратить значительное время на качественное обучение модели.</p>

<p>Угрозы:</p> <p>У1. Недостаток человеческих ресурсов по сравнению с конкурентными разработками.</p> <p>У2. В дальнейшем, используемых ресурсов для обучения может быть недостаточно</p> <p>У3: Высокая стоимость получения новых релевантных исходных данных.</p>	<p>У1У2С2:</p> <p>Недостаток ресурсов для обеспечения масштабирования проекта, дообучения модели</p>	<p>У1У2Сл2:</p> <p>В случае прекращения исследований по данной теме, разработка не сможет быть реализована в полной мере, не обеспечит улучшения и продвижения в данном направлении среди конкурентов.</p>
---	--	--

На основе SWOT-анализа можно сделать вывод, что возможные проблемы исследовательского проекта могут быть решены с помощью его сильных сторон, в том числе улучшение технологий на основе данных, полученных при анализе конкурентов.

#### 4.1.4. Оценка готовности проекта к коммерциализации

Для исследовательских проектов крайне важно оценить готовность к коммерциализации и уровень знаний, необходимых для реализации. По результатам такого анализа можно сделать вывод о готовности исследовательского проекта к коммерциализации, а также о необходимости увеличения количества членов команды проекта.

Для оценки готовности научной разработки к коммерциализации была составлена таблица 5, содержащая показатели о степени проработанности проекта с позиции коммерциализации и компетенциям разработчика научного проекта.

Таблица 5 – Оценка степени готовности научного проекта к коммерциализации

№ п/п	Наименование	Степень проработанности научного проекта	Уровень имеющихся знаний у разработчика
1	Определён имеющийся научно-технический задел	4	5

2	Определены перспективные направления коммерциализации научно-технического задела	3	3
3	Определены отрасли и технологии (товары, услуги) для представления на рынок	3	3
4	Определена товарная форма научно-технического задела для представления на рынок	2	2
5	Определены авторы и осуществлена охрана их прав	4	3
6	Проведена оценка стоимости интеллектуальной собственности	3	3
7	Проведены маркетинговые исследования рынков сбыта	3	3
8	Разработан бизнес-план коммерциализации научной разработки	4	3
9	Определены пути продвижения научной разработки на рынок	2	2
10	Разработана стратегия реализации научной разработки	4	5
11	Проработаны вопросы международного сотрудничества и выхода на зарубежный рынок	2	2
12	Проработаны вопросы использования инфраструктуры поддержки, получения льгот	2	1
13	Проработаны вопросы финансирования коммерциализации научной разработки	3	3
14	Имеется команда для коммерциализации научной разработки	5	5
15	Проработан механизм реализации научной разработки	4	4
<b>ИТОГО</b>		<b>48</b>	<b>47</b>

Итоговые результаты показывают проработанность проекта выше среднего. Для успешного дальнейшего продвижения необходимо развить компетенции в области экономики и маркетинга.

#### **4.1.5. Методы коммерциализации результатов научно-технического исследования**

Когда речь идет о коммерциализации научно-технических разработок, продавец (обычно владелец соответствующей интеллектуальной собственности) преследует четкие цели, которые в значительной степени зависят от того, куда он намерен направить (использовать, вложить) полученный коммерческий эффект. Целю коммерциализации может быть получение средств для продолжения научных исследований и разработок, одноразовое получение финансовых ресурсов, обеспечение постоянного притока финансовых средств.

Срок продвижения товара на рынок во многом зависит от того, какой метод коммерциализации был выбран. Этот раздел посвящен выбору метода коммерциализации объекта исследования и обоснованию его целесообразности.

В данной работе возможны следующие методы коммерциализации научных разработок:

1. Торговля патентными лицензиями, т.е. передача третьим лицам права использования объектов интеллектуальной собственности на лицензионной основе. При этом в патентном законодательстве выделяющие виды лицензий: исключительные (простые), исключительные, полные лицензии, сублицензии, опционы.
2. Инжиниринг как самостоятельный вид коммерческих операций предполагает предоставление на основе договора инжиниринга одной стороной, именуемой консультантом, другой стороне, именуемой заказчиком, комплекса или отдельных видов инженерно-технических услуг, связанных с проектированием, строительством и вводом объекта в эксплуатацию, с разработкой новых технологических процессов на предприятии заказчика, усовершенствованием имеющихся производственных процессов вплоть до внедрения изделия в производство и даже сбыта продукции.

### 3. Передача интеллектуальной собственности в уставной капитал предприятия.

В рамках данной работы инжиниринг является наиболее подходящим способом коммерциализации, так как использование ресурсов заказчика упрощает и ускоряет разработку технологического решения.

#### **4.2. Инициация проекта**

Группа процессов инициации состоит из процессов, которые выполняются для определения нового проекта или новой фазы существующего. В рамках этих процессов определяются изначальные цели и содержание и фиксируются изначальные финансовые ресурсы. Определяются внутренние и внешние заинтересованные стороны проекта, которые будут взаимодействовать и влиять на общий результат научного проекта. Данная информация закрепляется в Уставе проекта.

Инициация проекта состоит из процессов, которые выполняются для нового проекта или новой стадии проекта. Для этого определяются начальные цели, содержание, фиксируются ресурсы. Также определяются внутренние и внешние заинтересованные стороны проекта. Эта информация закрепляется в уставе проекта.

Устав проекта документирует бизнес-потребности, текущее понимание потребностей заказчика проекта, а также новый продукт, услугу или результат, который планируется создать.

##### **4.2.1. Цели и результат проекта**

В данном разделе приведена информация о заинтересованных сторонах проекта, иерархии целей проекта и критериях достижения целей.

Под заинтересованными сторонами проекта понимаются лица или организации, которые активно участвуют в проекте или интересы которых могут быть затронуты как положительно, так и отрицательно в ходе исполнения или в результате завершения проекта. Это могут быть заказчики,



спонсоры, общественность и т.п. Информация по заинтересованным сторонам проекта представлена в таблице 6.

Таблица 6 – Заинтересованные стороны проекта

<b>Заинтересованные стороны проекта</b>	<b>Ожидания заинтересованных сторон</b>
Научный руководитель, магистрант	Завершение выпускной квалификационной работы.
Компания-пользователь	Получение результатов обнаружения болезней с/х культур на изображениях, разработка программного обеспечения с использованием моделей нейронных сетей
Разработчик	Получение опыта разработки, получение заработной платы.
Компания разработчика	Разработка нового программного обеспечения, получение прибыли и инвестиций.

Цели и результаты проекта представлены в таблице 7

Таблица 7 – Цели и результаты проекта

<b>Цели проекта:</b>	<ul style="list-style-type: none"> <li>• Осуществить обзор существующих методов и алгоритмов обнаружения болезней сельскохозяйственных культур.</li> <li>• Подготовить набор больших данных, связанных со здоровьем сельскохозяйственных культур.</li> <li>• Реализовать алгоритм обнаружения болезней, который может эффективно обрабатывать большие данные.</li> <li>• Обучить модель на больших данных и оценить ее производительность.</li> <li>• Разработать на основе обученной модели сервис обнаружения болезней.</li> <li>• Представить результаты исследования, выводы и предложения.</li> </ul>
<b>Ожидаемые результаты проекта:</b>	<ul style="list-style-type: none"> <li>• Был разработан сервис обнаружения болезней сельскохозяйственных культур на основе нейросетевой модели архитектуры EfficientNet.</li> <li>• Выпускная квалификационная работа завершена.</li> </ul>
<b>Критерии приёмки результата проекта:</b>	<ul style="list-style-type: none"> <li>• Успешное тестирование на тестовых выборках данных;</li> <li>• Удовлетворительные значения метрик точности классификации;</li> <li>• Удовлетворительная скорость вычислений модели СНС.</li> </ul>

<b>Требования к результату проекта:</b>	Разработанный сервис должен быть основан на модели нейронной сети, которая обучается на большом объеме данных.
	Точность обнаружения болезней у разработанного алгоритма должна быть выше (97%), чем у большинства конкурентных решений
	Вес модели нейронной сети в МБ должны быть, по возможности, минимальными.

#### 4.2.2. Организационная структура проекта

При проведении сложных научно-исследовательских работ необходимо организовать группу способную разработать сам проект, оценить возможные экономические и социальные риски проекта и удовлетворить требования потенциальных потребителей.

В таблице 8 отразим организационную структуру, роль и функции каждого члена команды.

Таблица 8 – Рабочая группа проект

<b>№ п/п</b>	<b>ФИО, основное место работы, должность</b>	<b>Роль в проекте</b>	<b>Функции</b>	<b>Трудозатраты, час.</b>
1	Ковин Роман Владимирович, к.т.н., Доцент Отделения информационных технологий, Инженерная школа информационных технологий и робототехники	Руководитель проекта	Постановка целей и задач; Разработка и мониторинг календарного плана; Проведение консультаций по целям и задачам выпускной квалификационной работы; Оценка эффективности полученных результатов.	140
2	Спицына Любовь Юрьевна, к.э.н., Доцент Отделения социально-гуманитарных наук, Школа базовой инженерной подготовки	Эксперт проекта	Консультирование по разделу выпускной квалификационной работы «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение».	48

3	Антоневич Ольга Алексеевна, к.б.н., Доцент Отделения общетехнических дисциплин, Школа базовой инженерной подготовки	Эксперт проекта	Консультирование по разделу выпускной квалификационной работы «Социальная ответственность».	40
4	Ануфриева Татьяна Николаевна, Старший преподаватель Отделения иностранных языков, Школа базовой инженерной подготовки	Эксперт проекта	Консультации по разделу на английском языке выпускной квалификационной работы.	16
5	Жиганов Владислав Вадимович, Магистрант	Исполнитель по проекту	Изучение предметной области; Подготовка набор больших данных изображений; Разработка сервиса обнаружения болезней сельскохозяйственных культур на основе нейросетевой модели; Составление пояснительной записки.	800
<b>Итого</b>				<b>1 044</b>

#### 4.2.3. Ограничения и допущения проекта

В таблице 9 отразим факторы, которые могут играть роль в ограничении степени свободы участников команды проекта.

Таблица 9 – Ограничения проекта

<b>Фактор</b>	<b>Ограничения / Допущения</b>
Источник финансирования	Средства ТПУ
Сроки реализации проекта	30.01.2023 — 27.06.2023
Дата утверждения плана проекта	03.03.2023
Дата завершения	27.06.2023
Продолжительность работы над проектом в день	8 часов

### 4.3. Планирование управления научно-техническим проектом

Группа процессов планирования включает процессы, предпринимаемые для определения общего объема работ, постановки целей и разработки ряда мероприятий, необходимых для достижения этих целей.

#### 4.3.1. Иерархическая структура работ проекта

Иерархическая структура работ (ИСР) – детализация укрупнённой структуры работ. В процессе создания ИСР структурируется и определяется содержание всего проекта. Диаграмма ИСР показана на рисунке 13.

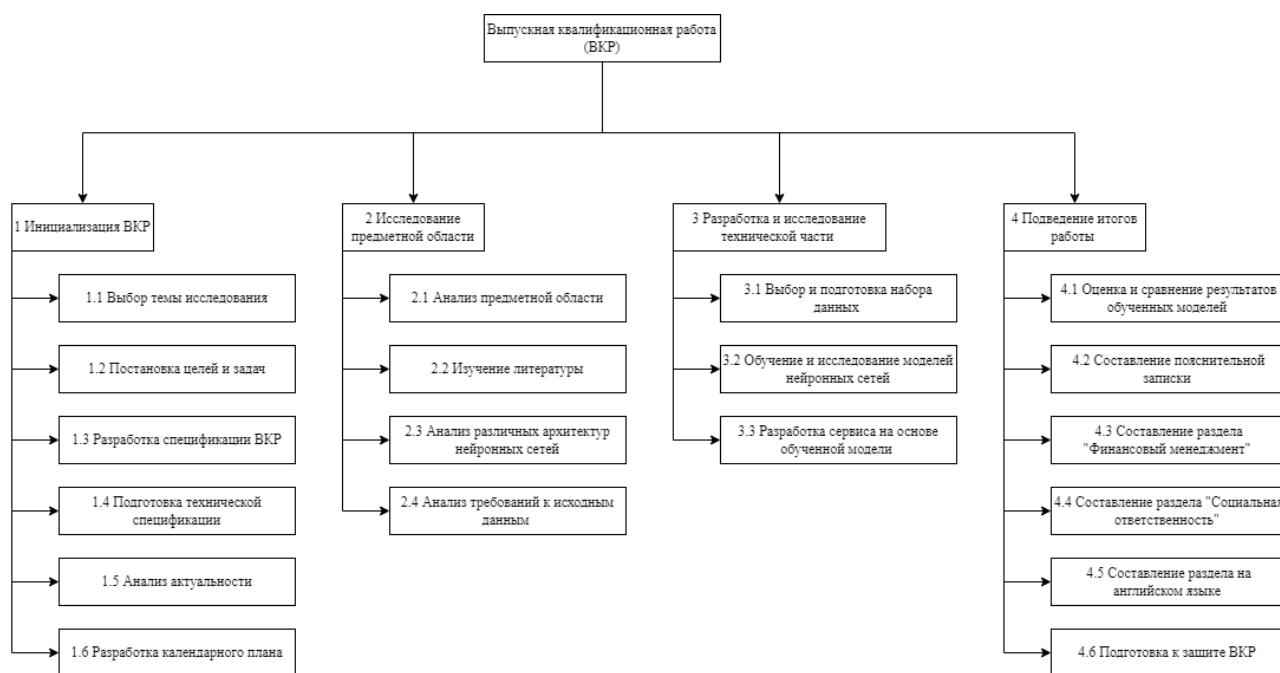


Рисунок 13 – Иерархическая структура работ

#### 4.3.2. План проекта

При организации работ по внедрению необходимо обоснованно распределить труд и рабочее время каждого участника процесса. Для этого был составлен полный перечень работ, представленный в таблице 9. Рабочие дни были рассчитаны исходя из шестидневной рабочей недели, с учётом праздничных дней Российской Федерации. Список участников представлен ниже:

- НР — Научный руководитель (Руководитель проекта), Ковин Роман Владимирович;

- ФМ — Эксперт по разделу Финансового Менеджмента, Спицына Любовь Юрьевна;
- СО — Эксперт по разделу Социальной Ответственности, Антоневиц Ольга Алексеевна;
- АЧ — Эксперт по разделу на английском языке, Ануфриева Татьяна Николаевна;
- С — Исполнитель по проекту (Магистрант), Жиганов Владислав Вадимович.

Таблица 10 – Календарный план проекта

№	Название	Длительность, раб. дни	Дата начала работ	Дата окончания работ	Состав участников
1	Инициация ВКР				
1.1	Выбор темы исследования	5	30.01.2023	03.02.2023	НР, С
1.2	Постановка целей и задач	3	04.02.2023	07.02.2023	НР
1.3	Разработка спецификации ВКР	6	08.02.2023	14.02.2023	НР, С
1.4	Подготовка технической спецификации	3	15.02.2023	17.02.2023	С
1.5	Анализ актуальности	7	18.02.2023	28.02.2023	С
1.6	Разработка календарного плана	3	01.03.2023	03.03.2023	НР, С
2	Исследование предметной области и технологий				
2.1	Анализ предметной области	9	04.03.2023	15.03.2023	С
2.2	Изучение литературы	9	16.03.2023	24.03.2023	С
2.3	Анализ различных архитектур нейронных сетей	6	25.03.2023	31.03.2023	С
2.4	Анализ требований к исходным данным	4	01.04.2023	05.04.2023	С

3	Разработка и исследование технической части				
3.1	Выбор и подготовка набора данных	10	06.04.2023	17.04.2023	С
3.2	Обучение и исследование моделей нейронных сетей	10	18.04.2023	28.04.2023	С
3.3	Разработка сервиса на основе обученной модели	10	29.04.2023	13.05.2023	С
4	Подведение итогов работы				
4.1	Оценка и сравнение результатов обученных моделей	3	14.05.2023	17.05.2023	НР, С
4.2	Составление пояснительной записки	16	18.05.2023	05.06.2023	С
4.3	Составление раздела "Финансовый менеджмент"	6	06.06.2023	13.06.2023	С, ФМ
4.4	Составление раздела "Социальная ответственность"	5	14.06.2023	19.06.2023	С, СО
4.5	Составление раздела на английском языке	2	20.06.2023	21.06.2023	С, АЧ
4.6	Подготовка к защите ВКР	5	22.06.2023	27.06.2023	НР, С

Диаграмма Ганта – это столбчатая диаграмма, показывающая ход выполнения проекта, с указанием дат начала и окончания работ, представляющая деятельность субъекта в виде отрезка времени. Диаграмма Ганта, построенная по рассчитанным значениям, представлена на рисунке 14.

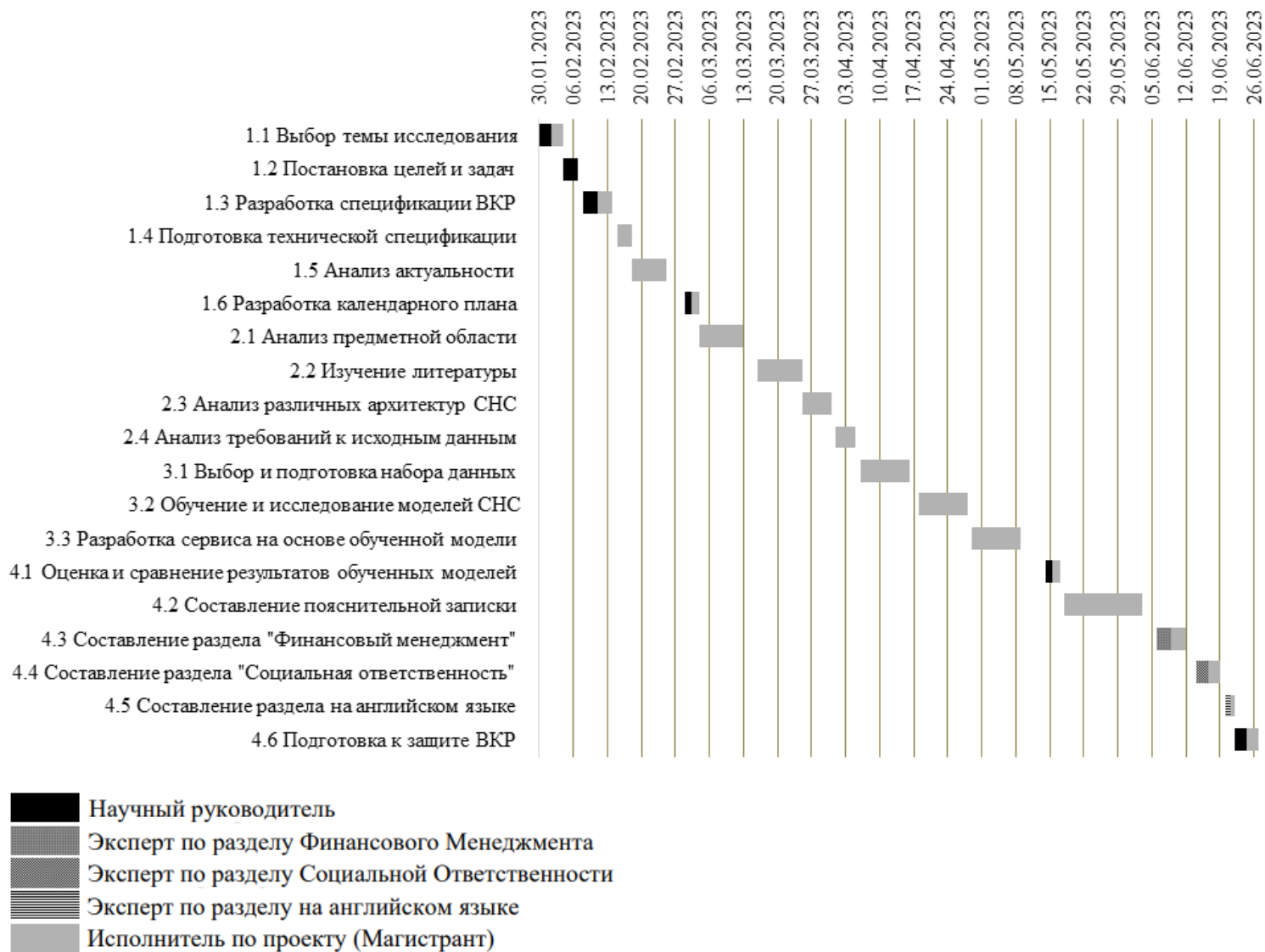


Рисунок 14 – Диаграмма Ганта по проекту

#### 4.4. Бюджет научного исследования

##### 4.4.1. Сырье, материалы, покупные изделия и полуфабрикаты

Разработка программного обеспечения осуществляется на персональном компьютере без использования материального сырья. Расчёт данной статьи расходов не требуется.

##### 4.4.2. Затраты на электроэнергию

В ходе выполнения работы потребляются энергетические ресурсы. Затраты на электроэнергию при работе оборудования для технологических целей рассчитывают по формуле:

$$C = M_э \times M_о \times И_о , \quad (5)$$

где  $M_э$  — тариф на электроэнергию (3,16 руб. за 1 кВт/ч [15]);

$M_о$  — мощность оборудования, кВт;

$И_о$  — время использования оборудования, часы;

Стоимость электроэнергии за 800 часов работы при использовании оборудования мощностью 0,4 кВт составляет:

$$C = 3,16 \times 0,4 \times 800 = 1\,011,2 \text{ руб} \quad (6)$$

##### 4.4.3. Специальное оборудование для научных (экспериментальных) работ

В качестве специального оборудования подразумеваются лицензии на ПО и затраты на приобретение ПК для разработки.

Таблица 11 – Расчёт стоимости специального оборудования

№ п/п	Наименование	Кол-во, шт.	Цена единицы, руб.	Общая стоимость, руб.
1	Персональный компьютер	1	125 000,00	125 000,00
<b>Итого:</b>				125 000,00



#### 4.4.4. Основная заработная плата

В данном разделе рассматриваются оклад, стимулирующие и надбавки. Расчёт основан на сложности каждого этапа, а также на месячной зарплате исполнителя. Расчёт основной заработной платы представлен в таблице 12.

Таблица 12 – Расчёт основной заработной платы

№ п/п	Наименование этапов	Исполнители по категориям	Трудоёмкость, чел.-дн.	Заработная плата, приходящаяся на один чел.-дн., руб.	Всего заработная плата по тарифу, руб.
1	Инициация ВКР	НР	12	4 627,79	55 533,48
2	Инициация ВКР	С	18	2 995,17	53 913,06
3	Исследование предметной области и технологий	С	28	2 995,17	83 864,76
4	Разработка и исследование технической части	С	30	2 995,17	89 855,10
5	Подведение итогов работы	НР	8	4 627,79	37 022,32
6	Подведение итогов работы	ФМ	6	4 627,79	27 766,74
7	Подведение итогов работы	СО	5	4 627,79	23 138,95
8	Подведение итогов работы	АЧ	2	4 627,79	9 255,58
9	Подведение итогов работы	С	37	2 995,17	110, 821,29
<b>Итого</b>					<b>398 615,48</b>

Статья включает в себя основную и дополнительную заработную плату работников, которые непосредственно заняты выполнением проекта, а также премии и доплаты:

$$C_{зп} = Z_{осн} + Z_{доп}, \quad (7)$$

Где  $Z_{осн}$  – основная заработная плата;

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

Основная заработная плата рассчитывается по следующей формуле:

$$Z_{осн} = Z_{дн} \times T_{раб}, \quad (8)$$

Где  $Z_{осн}$  – основная заработная плата одного работника, руб.;

$T_p$  – продолжительность работ, выполняемых научно-техническим работником, раб. дн.;

$Z_{дн}$  – среднедневная заработная плата работника, руб.

Среднедневная заработная плата рассчитывается по формуле:

$$Z_{дн} = \frac{Z_m \times M}{F_d}, \quad (9)$$

Где  $Z_m$  – месячный должностной оклад работника, руб.;

$M$  – количество месяцев работы без отпуска в течение года: при отпуске в 48 раб. дня  $M = 10,4$  месяца, 6-дневная неделя;

$F_d$  – действительный годовой фонд рабочего времени научно-технического персонала, рабочие дни (252).

В таблице 13 показано количество календарных дней, нерабочих и праздничных дней, дней в связи с потерей рабочего времени и фактический годовой фонд рабочего времени.

Таблица 13 – Баланс рабочего времени

Показатели рабочего времени	Работник
Календарное число дней	365
Количество нерабочих дней	
– выходные дни	54
– праздничные дни	11
Потери рабочего времени	
– отпуск	48
– невыходы по болезни	0
Действительный годовой фонд рабочего времени	252

Месячный должностной оклад работника рассчитывается по формуле:

$$Z_m = Z_b \times (k_{пр} + k_d) \times k_p, \quad (10)$$

Где  $Z_b$  — базовый оклад, руб.;

$k_{пр}$  — премиальный коэффициент;

$k_d$  — коэффициент доплат и надбавок;

$k_p$  — районный коэффициент, равный 1,3 для Томска.

Расчёт основной заработной платы приведён в таблице 14.

Таблица 14 – Расчёт основной заработной платы

Исполнители	$Z_b$ , руб.	$k_{пр}$	$k_d$	$k_p$	$Z_m$ , руб.	$Z_{дн}$ , руб.	$T_{раб}$ , дн.	$Z_{осн}$ , руб.
Научный руководитель	37 700	1,1	1,188	1,3	112 134,88	4 627,79	20	92 555,8
Эксперт по разделу Финансового Менеджмента	37 700				112 134,88	4 627,79	6	27 766,74
Эксперт по разделу Социальной Ответственности	37 700				112 134,88	4 627,79	5	23 138,95
Эксперт по английской части	37 700				112 134,88	4 627,79	2	9 255,58
Исполнитель по проекту (Магистрант)	24 400				72 575,36	2 995,17	113	338 454,21

#### 4.4.5. Дополнительная заработная плата

В данную статью включается сумма выплат, предусмотренных законодательством о труде, например, оплата очередных и дополнительных отпусков; оплата времени, связанного с выполнением государственных и общественных обязанностей; выплата вознаграждения за выслугу лет и т.п.

Дополнительная заработная плата рассчитывается исходя из 10-15% от основной заработной платы работников по следующей формуле:

$$Z_{доп} = Z_{осн} \times k_{доп}, \quad (11)$$

Где  $Z_{доп}$  — дополнительная заработная плата, руб.;

$Z_{\text{осн}}$  — основная заработная плата, руб.;

$k_{\text{доп}}$  — коэффициент дополнительной зарплаты (10%).

В таблице 15 представлен расчёт дополнительной заработной платы.

Таблица 15 – Расчёт дополнительной заработной платы

Исполнители	Основная зарплата, руб.	Дополнительная зарплата, руб.	Зар. плата исполнителя, руб.
Научный руководитель	92 555,8	9 255,58	101 811,38
Эксперт по разделу Финансового Менеджмента	27 766,74	2 776,67	30 543,41
Эксперт по разделу Социальной Ответственности	23 138,95	2 313,90	25 452,85
Эксперт по английской части	9 255,58	925,60	10 181,14
Исполнитель по проекту (Магистрант)	338 454,21	33 845,42	372 299,63

Таким образом, общая сумма дополнительной заработной платы равна 39 861,59 руб.

#### 4.4.6. Отчисления на социальные нужды

Отчисления на социальное страхование (так называемый трудовой налог) во внебюджетные фонды являются обязательными по нормам, установленным законодательством Российской Федерации на государственное социальное страхование, пенсионный фонд и медицинское страхование от расходов работников.

Платёж во внебюджетные фонды определяется по формуле:

$$C_{\text{внеб}} = k_{\text{внеб}} \times (Z_{\text{осн}} + Z_{\text{доп}}), \quad (12)$$

Где  $k_{\text{внеб}}$  — коэффициент отчислений на уплату во внебюджетные фонды.

В таблице 16 приведён расчёт отчислений на социальные нужды с коэффициентом отчислений, равным 30%.

Таблица 16 – Расчёт отчислений на социальные нужды

<b>Исполнители</b>	<b>Основная зарплата, руб.</b>	<b>Дополнительная зарплата, руб.</b>	<b>Отчисления, руб.</b>
Научный руководитель	92 555,8	9 255,58	30 543,41
Эксперт по разделу Финансового Менеджмента	27 766,74	2 776,67	9 163,02
Эксперт по разделу Социальной Ответственности	23 138,95	2 313,90	7 635,86
Эксперт по английской части	9 255,58	925,60	30 544,67
Исполнитель по проекту (Магистрант)	338 454,21	33 845,42	111 689,89

Таким образом, общая сумма отчислений на социальные нужды равна 189 576,85 руб.

#### **4.4.7. Накладные расходы**

В эту статью включаются затраты на управление и хозяйственное обслуживание, которые могут быть отнесены непосредственно на конкретную тему. Кроме того, сюда относятся расходы по содержанию, эксплуатации и ремонту оборудования, производственного инструмента и инвентаря, зданий, сооружений и др. В расчётах эти расходы принимаются в размере 70-90% от суммы основной заработной платы научно-производственного персонала данной научно-технической организации.

Накладные расходы составляют 16% от суммы основной и дополнительной заработной платы, работников, непосредственно участвующих в выполнение темы.

Расчёт накладных расходов ведётся по следующей формуле:

$$C_{\text{накл}} = k_{\text{накл}} \times (Z_{\text{осн}} + Z_{\text{доп}}), \quad (13)$$

Где  $k_{\text{накл}}$  — коэффициент накладных расходов.

В таблице 17 представлен расчёт накладных расходов.

Таблица 17 – Расчёт накладных расходов

<b>Исполнители</b>	<b>Основная зарплата, руб.</b>	<b>Дополнительная зарплата, руб.</b>	<b>Накладные расходы, руб.</b>
--------------------	--------------------------------	--------------------------------------	--------------------------------

Научный руководитель	92 555,8	9 255,58	16 289,82
Эксперт по разделу Финансового Менеджмента	27 766,74	2 776,67	4 886,95
Эксперт по разделу Социальной Ответственности	23 138,95	2 313,90	4 072,46
Эксперт по английской части	9 255,58	925,60	1 628,98
Исполнитель по проекту (Магистрант)	338 454,21	33 845,42	59 567,94

Таким образом, общая сумма накладных расходов равна 86 446,15 руб.

#### 4.4.8. Формирование бюджетных расходов

Результаты формирования бюджетных расходов представлены в таблице 18.

Таблица 18 – Бюджет затрат на проект

Статьи	Стоимость, руб.
Расходы на электроэнергию	1 011,20
Расходы на специальное оборудование для научных экспериментов	125 000,00
Основная заработная плата	398 615,48
Дополнительная зарплата	39 861,59
Социальные отчисления	189 576,85
Накладные расходы	86 446,15
Итого запланированные расходы	840 511,27

#### 4.5. Оценка сравнительной эффективности исследования

Определение эффективности происходит на основе расчёта интегрального показателя эффективности научного исследования. Его нахождение связано с определением двух средневзвешенных величин: финансовой эффективности и ресурсоэффективности.

Интегральный финансовый показатель разработки определяется как:

$$I_{\Phi}^p = \frac{\Phi_{pi}}{\Phi_{\max}}, \quad (14)$$

Где  $I_{\Phi}^p$  — интегральный финансовый показатель разработки;

$\Phi_{pi}$  — стоимость -го варианта исполнения;

$\Phi_{max}$  — максимальная стоимость исполнения научно-исследовательского проекта (в т.ч. аналоги).

Интегральный показатель ресурсоэффективности вариантов исполнения объекта исследования можно определить следующим образом:

$$I_m^a = \sum_{i=1}^n a_i \times b_i^a, \quad (15)$$

$$I_m^p = \sum_{i=1}^n a_i \times b_i^p, \quad (16)$$

Где  $I_m^a, I_m^p$  — интегральный показатель ресурсоэффективности вариантов;  
 $a_i$  — весовой коэффициент  $i$ -го параметра;

$b_i^a, b_i^p$  — бальная оценка  $i$ -го параметра для аналога и разработки, устанавливается экспертным путём по выбранной шкале оценивания;

$n$  — число параметров сравнения.

В качестве параметров сравнения были взяты критерии конкурентоспособности из подраздела 1.1.2 с теми же весовыми коэффициентами. Результаты определения интегрального показателя ресурсоэффективности приведены в таблице 19. Под аналогом 1 подразумевается GoogLeNet, под аналогом 2 — AlexNet из подраздела 1.1.2.

Таблица 19 – Сравнительная оценка характеристик вариантов исполнения проекта

Критерии	Весовой коэффициент параметра	Текущий проект	Аналог 1	Аналог 2
Точность	0,25	5	4	3
Требование к вычислительным ресурсам	0,10	4	5	5
Возможность дообучения	0,05	4	3	3
Универсальность	0,15	5	2	1
Востребованность	0,10	4	3	2

Требовательность к данным	0,15	5	3	2
Стоимость разработки	0,10	5	4	3
Ожидаемый жизненный цикл	0,10	5	2	4
<b>Итого</b>	1	4,75	3,30	2,75

Интегральный показатель эффективности разработки ( $I_{\text{финр}}^p$ ) и аналога ( $I_{\text{финр}}^a$ ) определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле:

$$I_{\text{финр}}^p = \frac{I_m^p}{I_{\Phi}^p}, \quad (17)$$

$$I_{\text{финр}}^a = \frac{I_m^a}{I_{\Phi}^a}, \quad (18)$$

Сравнение интегрального показателя эффективности текущего проекта и аналогов позволит определить сравнительную эффективность проекта. Сравнительная эффективность проекта определяется по формуле:

$$\mathcal{E}_{\text{ср}} = \frac{I_{\text{финр}}^p}{I_{\text{финр}}^a}, \quad (19)$$

Где  $\mathcal{E}_{\text{ср}}$  — сравнительная эффективность проекта;

$I_{\text{финр}}^p$  — интегральный показатель разработки;

$I_{\text{финр}}^a$  — интегральный технико-экономический показатель аналога.

Результаты расчётов представлены в таблице 20.

Таблица 20 – Сравнительная эффективность исследования

Показатели	Текущий проект	Аналог 1	Аналог 2
Интегральный финансовый показатель разработки	1	0,695	0,579
Интегральный показатель ресурсоэффективности разработки	4,75	3,30	2,75
Интегральный показатель эффективности	4,75	4,748	4,75
Сравнительная эффективность вариантов исполнения	1	1,0004	1

Сравнение значений интегральных показателей эффективности показывает, что текущий проект и аналоги схожи по эффективности, но



аналоги являются менее точными, менее быстрыми и более зависимыми от входных данных.

#### **4.6. Выводы по разделу**

Таким образом, в данном разделе были рассмотрены этапы проектирования и создания конкурентоспособных разработок, отвечающих требованиям ресурсоэффективности и ресурсосбережения.

Целью данной работы является разработка сервиса, способного автоматически выявлять заболевания сельскохозяйственных культур на изображениях.

Была построена карта потенциальных потребителей. Выполнен SWOT-анализ, который описывает сильные и слабые стороны проекта, а также возможности и угрозы для проекта. Была проведена оценка готовности проекта к коммерциализации, перспективы развития были признаны выше среднего. Для повышения оценки готовности проекта к коммерциализации необходимо провести консультации с экспертами в сферах, затрагиваемых разработкой, сформировать команду и найти финансирование.

На этапе инициации проекта определяются внутренние и внешние заинтересованные стороны проекта, а также цели и ожидаемые результаты проекта. График проекта отображается на диаграмме Ганта.

Рассчитана основная и дополнительная заработная плата исполнителей проекта. Бюджет проекта составил 840 511,27 руб. после вычетов на социальные нужды и накладные расходы.

Определена сравнительная эффективность проекта, которая, согласно расчётам, не уступает аналогам.

## **ГЛАВА 5. СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ**

Целью данной работы является создание сервиса, способного автоматически выявлять заболевания сельскохозяйственных культур на изображениях.

Разработка сервиса обнаружения болезней сельскохозяйственных культур представляет собой создание программного продукта, который позволит определять наличие заболеваний у растений на основе анализа изображений с использованием архитектуры нейронных сетей EfficientNet. Для обучения моделей СНС используется набор данных, содержащий изображения здоровых и больных растений сельскохозяйственных культур. Результатом вычисления модели нейронной сети для каждого изображения является определение наличия или отсутствия заболевания и его типа.

Разработанный сервис может быть использован на предприятиях сельского хозяйства, занимающихся полеводством, овощеводством и садоводством, для раннего выявления заболеваний.

В данном разделе проведён анализ опасных и вредных факторов труда, определён комплекс мер организационного, технического и правового характера, который должен препятствовать возникновению или снижению возможности негативных последствий в процессе осуществления работы разработчика программного обеспечения.

Рабочей зоной при разработке сервиса является рабочее место для выполнения работ сидя с использованием ПЭВМ.

### **5.1. Правовые и организационные вопросы обеспечения безопасности**

#### **5.1.1. Специальные правовые нормы трудового законодательства**

Правовые нормы устанавливаются специализированными государственными органами. Основные требования содержатся в Трудовом кодексе Российской Федерации.

Работа в офисе относится ко второй категории тяжести труда – работы выполняются при оптимальных условиях внешней производственной среды и при оптимальной величине физической, умственной и нервно-эмоциональной

нагрузки. Продолжительность рабочего дня работников не должна превышать 40 часов в неделю. Продолжительность непрерывной работы за компьютером без регламентированного перерыва не должна превышать 1 час. В течение рабочего дня работнику должен быть предоставлен перерыв для отдыха и питания продолжительностью не более двух часов и не менее 30 минут. Всем работникам предоставляются выходные дни (еженедельный непрерывный отдых). При пятидневной рабочей неделе работникам предоставляются два выходных дня в неделю, при шестидневной рабочей неделе – один выходной день.

Организация обязана предоставлять ежегодные отпуска с сохранением места работы (должности) и среднего заработка, продолжительностью 28 календарных дней. Для работников, занятых на работах с опасными или вредными условиями, предусматривается дополнительный отпуск.

#### **5.1.2. Организационные мероприятия при компоновке рабочей зоны**

Согласно общим принципам организации рабочего места, основными компонентами рабочего места программиста являются стол, стул, системный блок, монитор, клавиатура и мышь. При организации элементов рабочего места необходимо учитывать:

1. рабочую позу человека-оператора;
2. пространство для размещения человека-оператора;
3. возможность обзора элементов рабочего места;
4. возможность обзора пространства за пределами рабочего места;
5. возможность ведения записей, размещения документации и материалов, используемых человеком-оператором. [16]

Для обеспечения безопасности и комфорта работников при выполнении ими трудовых операций необходимо соблюдение следующих эргономических требований к рабочим местам: высота рабочей поверхности должна составлять 655 мм, высота сиденья – 430 мм, пространство для ног – не менее 500 мм. Подставка для ног должна регулироваться по высоте, ее ширина должна быть не менее 300 мм, длина – не менее 400 мм [17].

Дисплей на рабочем месте должен быть установлен ниже уровня глаз оператора. Угол наблюдения экрана оператором относительно горизонтальной линии взгляда не должен превышать  $60^\circ$ . Дисплей на рабочем месте оператора должен располагаться так, чтобы изображение в любой его части было различимо без необходимости поднять или опустить голову [18].

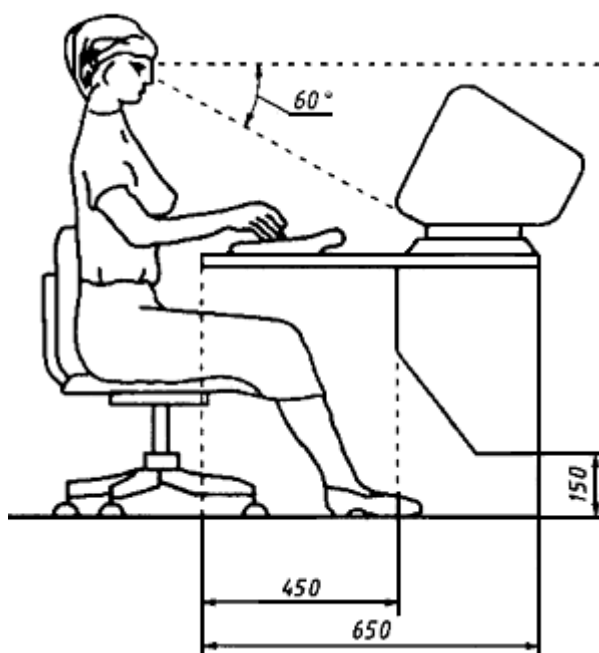


Рисунок 15 – Расположение дисплея на столе

## 5.2. Производственная безопасность

Человек подвергается воздействию опасностей в своей трудовой деятельности, которая осуществляется в пространстве, называемом производственной средой. В производственной среде объективно складываются вредные и опасные факторы, негативно воздействующие на человека в процессе его жизнедеятельности. Производственный фактор считается вредным, если он потенциально может вызвать заболевание у работника, если же его воздействие может привести к травме, то такой фактор считается опасным. В таблице 21 приведены возможные опасные и вредные факторы.

Таблица 21 – Опасные и вредные производственные факторы

Факторы (ГОСТ 12.0.003-2015)	Нормативные документы
Производственные факторы, связанные с недостаточной освещённостью рабочей зоны.	СП 52.13330.2016 Естественное и искусственное освещение. Актуализированная редакция СНиП 23-05-95 [19];
Производственные факторы, связанные с аномальными микроклиматическими параметрами воздушной среды на местонахождении работающего	СанПиН 1.2.3685-21 "Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания" [20]; ГОСТ 12.1.005-88 ССБТ. Общие санитарно-гигиенические требования к воздуху рабочей зоны [21];
Производственные факторы, связанные с акустическими колебаниями в производственной среде и характеризующиеся неблагоприятными характеристиками шума	СП 51.13330.2011 Защита от шума. Актуализированная редакция СНиП 23-03-2003 [22];
Производственные факторы, связанные с электромагнитными полями, неионизирующими ткани тела человека постоянного характера, связанного с повышенным образованием электростатических зарядов	ГОСТ 12.1.006-84 «Система стандартов безопасности труда (ССБТ). Электромагнитные поля радиочастот. Допустимые уровни на рабочих местах и требования к проведению контроля» [23];
Производственные факторы, связанные с перенапряжением анализаторов, вызванные информационной нагрузкой	МР 2.2.9.2311-07 «Профилактика стрессового состояния работников при различных видах профессиональной деятельности» [24];
Производственные факторы, связанные со статическими физическими перегрузками (нахождение в позе «сидя» или	ГОСТ 12.2.032-78 Система стандартов безопасности труда (ССБТ). Рабочее место при выполнении работ сидя. Общие

Факторы (ГОСТ 12.0.003-2015)	Нормативные документы
пребывание в вынужденной позе)	эргономические требования [17]; ГОСТ 22269-76 Рабочее место оператора. Взаимное расположение элементов рабочего места [16];

### 5.2.1. Производственные факторы, связанные с недостаточной освещённостью рабочей зоны.

Правильно спроектированное и выполненное производственное освещение улучшает условия зрительной работы, снижает утомляемость, способствует повышению производительности труда, благотворно влияет на производственную среду, оказывая положительное психологическое воздействие на работающего, повышает безопасность труда и снижает травматизм.

Недостаточность освещения приводит к напряжению зрения, ослабляет внимание, приводит к наступлению преждевременной утомленности. Чрезмерно яркое освещение вызывает ослепление, раздражение и резь в глазах. Неправильное направление света на рабочем месте может создавать резкие тени, блики, дезориентировать работающего.

Согласно СП 52.13330.2016 [19] при работах III зрительного разряда и подразряда г (работы высокой точности) освещённость при системе общего освещения должна быть не ниже 200 Лк.

Растёт общего равномерного искусственного освещения горизонтальной рабочей поверхности выполняется методом коэффициента использования светового потока, учитывающим световой поток, отражённый от потолка и стен. Длина помещения  $A = 4,0$  м, ширина  $B = 3,5$  м, высота  $H = 2,7$  м. Высота рабочей поверхности  $h_{рп} = 0,75$  м, требуемая освещённость  $E_n = 200$  Лк.

Размер помещения составляет:

$$S = A \times B = 4 \times 3,5 = 14 \text{ м}^2 \quad (20)$$

Стены помещения оклеены обоями светлых тонов, поэтому отражающая способность стен  $\rho_{ст} = 30\%$ , потолок белый матовый натяжной, что приравнивает его к побеленному, коэффициент отражения которого  $\rho_n = 50\%$ .

Для нормальных помещений с хорошим отражением потолка и стен выбираются открытые двухламповые светильники типа ОД, ОДОР, ШОД, ОДО или ООД. Так как высота помещения 2,7 м, то в соответствии с наименьшей допустимой высотой подвеса над полом необходимо выбрать тип светильника ШОД. Наилучшее расположение светильников данного типа берется с критерием  $\lambda = 1,1- 1,3$ . Световой поток одной лампы  $\Phi = 2300$  лм. Расстояние светильников от перекрытия  $h_c = 0,1$  м.

Высота светильника над полом определяется по формуле:

$$h_{п} = H - h_c = 2,7 - 0,1 = 2,6 \text{ м} \quad (21)$$

Из методической таблицы следует, что наименьшая допустимая высота подвеса над полом для типов светильников ОД при одиночной установке является 3,5 м., что не соответствует фактическому значению.

Тогда расчётная высота светильника над рабочей поверхностью вычисляется по формуле:

$$h = h_{п} - h_{рп} = 2,7 - 0,75 = 1,95 \text{ м} \quad (22)$$

Расстояние между светильниками определяется по формуле:

$$L = \lambda * h = 1,4 * 1,95 = 2,73 \text{ м} \quad (23)$$

Расстояние от крайнего ряда светильников до стены:

$$l = \frac{L}{3} = \frac{2,73}{3} = 0,91 \text{ м} \quad (24)$$

Количество рядов светильников с люминесцентными лампами определяется по формуле:

$$n_{ряд} = \frac{\left(B - \frac{2}{3} \times L\right)}{L} + 1 = \frac{\left(3,5 - \frac{2}{3} \times 2,73\right)}{2,73} + 1 \approx 1,615 = 2 \quad (25)$$

Количество светильников в ряду:

$$n_{св} = \frac{\left(A - \frac{2}{3} \times L\right)}{l_{св} + 0,5} = \frac{\left(4 - \frac{2}{3} \times 2,73\right)}{1,228 + 0,5} \approx 1,261 = 1 \quad (26)$$

Общее количество светильников с люминесцентными лампами в помещении определяется по формуле:

$$N = n_{\text{ряд}} \times n_{\text{св}} = 2 \times 1 = 2 \quad (27)$$

Учитывая, что в каждом светильнике установлено две лампы, общее число ламп в помещении  $N_{\text{л}} = 4$ .

Индекс помещения определяется по формуле:

$$i = \frac{S}{h \times (A + B)} = \frac{14}{1,95 \times (4 + 3,5)} \approx 0,957 = 1 \quad (28)$$

По таблице коэффициентов использования светового потока, для светильников типа ОД с люминесцентными лампами при  $\rho_n = 50\%$ ,  $\rho_{\text{ст}} = 30\%$  и индексе помещения  $i = 1$  равен  $\eta = 44\% = 0,44$ .

Из методической таблицы коэффициент запаса для помещений с малым выделением пыли  $K_3 = 1,5$ , коэффициент неравномерности  $Z = 1,1$ .

Световой поток ламп в каждом из рядов определяется по формуле:

$$\Phi_{\text{расч}} = \frac{E_{\text{н}} \times S \times K_3 \times Z}{N_{\text{л}} \times \eta} = \frac{200 \times 14 \times 1,5 \times 1,1}{4 \times 0,44} = 2625 \text{ лм} \quad (29)$$

Выбираем ближайшую стандартную лампу – ЛД 40 Вт с потоком 2300 Лм. Делаем проверку выполнения условия:

$$\begin{aligned} -10\% &\leq \frac{\Phi_{\text{стан}} - \Phi_{\text{расч}}}{\Phi_{\text{стан}}} \times 100\% \leq +20\% = \\ &= -10\% \leq \frac{2300 - 2625}{2300} \times 100\% \leq +20\% = \\ &= -10\% \leq -0,14\% \leq +20\% \end{aligned} \quad (30)$$

Таким образом:  $-10\% \leq -0,14\% \leq +20\%$ , необходимый световой поток светильника не выходит за пределы требуемого диапазона

Номинальная мощность осветительной системы рассчитывается по формуле:

$$P = N_{\text{л}} \times P_{\text{л}} = 4 \times 40 = 160 \text{ Вт} \quad (31)$$

Таким образом световой поток светильника не выходит за пределы требуемого диапазона.



### 5.2.2. Производственные факторы, связанные с аномальными микроклиматическими параметрами воздушной среды на местонахождении работающего

Одним из необходимых благоприятных условий труда является обеспечение нормального микроклимата в помещениях, что в свою очередь влияет на здоровье работников. Длительное воздействие неблагоприятных условий на человека ухудшает его самочувствие, снижает производительность труда и приводит к заболеваниям.

Согласно СанПиН 1.2.3685-21 [20] показатели температуры, относительной влажности и скорости движения воздуха оказывают существенное влияние на самочувствие человека.

В соответствии с ГОСТ 12.1.005-88 [21], оптимальные и допустимые показатели температуры, относительной влажности и скорости движения воздуха в рабочей зоне производственных помещений должны соответствовать значениям, указанным в таблице 22.

Таблица 22 – Оптимальные величины параметров микроклимата на рабочих местах в помещениях

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

Вычислительная техника является источником существенных тепловыделений, что может привести к повышению температуры и снижению относительной влажности в помещении. Для поддержания оптимальных показателей микроклимата используются системы отопления и кондиционирования воздуха, тепловая изоляция нагретых поверхностей оборудования. Исследования микроклимата показали, что параметры в помещениях, где проводились работы, соответствуют требованиям СанПиН.

### **5.2.3. Производственные факторы, связанные с акустическими колебаниями в производственной среде и характеризующиеся неблагоприятными характеристиками шума**

Шум является совокупностью звуков различной частоты, интенсивности и продолжительности. Высокий уровень шума, создаваемый печатными устройствами, множительной техникой, оборудованием для кондиционирования воздуха, вентиляторами систем охлаждения и трансформаторы в самих ЭВМ является одним из неблагоприятных факторов производственной среды программиста.

Длительное действие шума высокой интенсивности приводит к патологиям слухового органа и негативно влияет на нервную систему. Шум приводит к быстрой утомляемости человека, что в свою очередь ведет к производственным ошибкам.

Согласно СП 51.13330.2011, допустимый уровень шума составляет 60 дБА для рабочих помещений, и 50 дБА для офисных помещений и научно-исследовательских организаций [22].

Для снижения уровня шума стены и потолок помещений, где установлены компьютеры, могут быть облицованы звукопоглощающими материалами. Шумящее оборудование, уровни шума которого превышают нормированные, должно находиться вне помещений.

На рабочем месте уровень шума является допустимым, так как не вызывает значительного беспокойства, и не влияет на психологическое состояние.

### **5.2.4. Производственные факторы, связанные с электромагнитными полями, неионизирующими ткани тела человека постоянного характера, связанного с повышенным образованием электростатических зарядов**

В повседневной жизни для людей не заметно воздействие электромагнитных излучений. Степень воздействия электромагнитных полей на организм человека зависит от расстояния до источника, диапазона частот

излучения, интенсивности воздействия, продолжительности, характера и режима облучения, размера облучаемой поверхности и особенностей организма.

Уровень электромагнитных излучений регулируется ГОСТ 12.1.006-84 [23]. В таблице 23 указаны допустимые уровни электромагнитного поля, создаваемым персональным компьютером.

Таблица 23 –Допустимые уровни ЭМП, создаваемых ПК

Наименование параметров		ВДУ
Напряженность электрического поля	в диапазоне частот 5 Гц - 2 кГц	25 В/м
	в диапазоне частот 2 кГц - 400 кГц	2,5 В/м
Плотность магнитного потока	в диапазоне частот 5 Гц - 2 кГц	250 нТл
	в диапазоне частот 2 кГц - 400 кГц	25 нТл
Напряженность электростатического поля		15 кВ/м

Основные излучающие электромагнитное поле части ПЭВМ – это системный блок и экран монитора.

На рабочем месте установлены ПЭВМ, оснащённые жидкокристаллическим монитором. Они излучают электромагнитные волны, которые не причиняют человеку вреда, даже при длительной работе.

#### **5.2.5. Производственные факторы, связанные с перенапряжением анализаторов, вызванные информационной нагрузкой**

Работа с ПЭВМ связана с обработкой большого количества информации. Анализ данных, инженерные исследования, расчеты и разработка программного обеспечения требуют высокого уровня концентрации. При работе с визуальной информацией напрягаются глаза, которые являются зрительными анализаторами человека. Расстояние расположения предмета постоянного визуального контроля не меняется в процессе работы, что вызывает усталость глазных мышц и ухудшение зрения. При длительных контактах с дисплеем, постоянного наблюдения схожей по структуре зрительной информации, человек начинает испытывать стресс.

Разработка ПО требует сосредоточенности и частого переключения между использованием нескольких программ одновременно. Для устранения накопленной усталости и нагрузки на организм человека необходимо выполнять комплекс упражнений на концентрацию внимания, комплекс упражнений на глаз, использовать методику психической саморегуляции [24].

### **5.2.6. Производственные факторы, связанные со статическими физическими перегрузками (нахождение в позе «сидя» или пребывание в вынужденной позе)**

В современном мире, почти каждая работа, так или иначе, связана с работой за компьютером. Разработчики программного обеспечения, инженеры и те, кто учится – все вынуждены проводить часы за ПЭВМ. При этом пользователь вынужден принимать одну и ту же позу в течение длительного времени, тем самым создавая в работе мышечного корсета статические перегрузки. Неудобная поза, нахождение центра тяжести в одном месте, постоянный наклон вперед могут вызвать боли в шее и спине.

Длительное (25 и более процентов времени смены) пребывание работников различных профессий в физиологически нерациональных рабочих позах является утомительным для организма из-за постоянной статической нагрузки на отдельные мышечные группы, в основном на группы мышц рук, шеи, плечевого пояса, поясничной области, что может явиться причиной возникновения патологических нарушений [24].

Для устранения накопленной усталости и нагрузки на организм человека необходимо выполнять комплекс физических упражнений на координацию движений [24].

### **5.3. Экологическая безопасность**

Один из источников загрязнения окружающей среды на этом рабочем месте – это воздействие на литосферу и атмосферу, вызванное образованием отходов при поломке компьютерной и оргтехники и неправильной утилизацией отходов. Нормативы экологической безопасности установлены ГОСТ 17.4.3.04-85 и ГОСТ Р 53692-2009 [25, 26].

Согласно ГОСТ Р 53692-2009 [26], вышедшее из строя ПЭВМ и сопутствующая оргтехника относится к IV классу опасности и подлежит специальной утилизации. Для оказания наименьшего влияния на окружающую среду, необходимо проводить специальную процедуру утилизации ПЭВМ и оргтехники, при которой более 90% отправится на вторичную переработку и менее 10% будут отправлены на свалки. При этом она должна соответствовать процедуре утилизации ГОСТ Р 53692-2009.

Первый этап – это утилизация инертных отходов. В процессе утилизации могут быть переработаны бракованные или вышедшие из употребления виды продукции, их части и отходы от них путем разборки, переплавки или других технологий с целью восстановления органических и неорганических компонентов.

На втором этапе отходы I-IV классов опасности размещаются на соответствующих полигонах или уничтожаются.

Вышедшие из строя люминесцентные лампы являются распространенным источником ртутного загрязнения.

Каждая лампа содержит около 60 мг ртути – чрезвычайно опасного химического вещества. Согласно СанПиН 1.2.3685-21 [20], ртуть относится к I классу опасности вредных веществ, предельно допустимая концентрация паров ртути в атмосферном воздухе составляет 0,0003 мг/м<sup>3</sup>. Превышение концентрации вредных веществ может вызвать отравление живых существ и загрязнение атмосферы, гидросферы и литосферы. После пятилетнего срока службы такие лампы необходимо сдавать на переработку в специальные пункты приема.

#### **5.4.Безопасность в чрезвычайных ситуациях**

Чрезвычайная ситуация (ЧС) складывается в результате аварии, опасного природного явления, катастрофы, распространения заболевания, стихийного или иного бедствия. Она представляет опасность для окружающих, поскольку может повлечь человеческие жертвы и значительный ущерб инфраструктуре, здоровью, условиям жизнедеятельности и окружающей среде.

Большинство ЧС носят техногенный характер, т.е. возникают вследствие аварии на техническом объекте. В качестве анализа мер безопасности при чрезвычайной ситуации была выбрана наиболее распространенная ЧС, а именно пожар.

Согласно ГОСТ 12.1.004-91 [27], помещение для работы операторов ПЭВМ относится к категории низкого пожарного риска (Д). Риск возникновения пожара возрастает в помещениях с ПК из-за ряда факторов, включая перегрев электронных компонентов, короткое замыкание и неисправность электропроводки.

Для предотвращения возникновения пожаров, в зависимости от их возможных причин, необходимо применить следующие меры по обеспечению противопожарной защиты:

- Необходимо проводить регулярные проверки первичных средств тушения пожаров и систем оповещения.
- Должны проводиться инструктажи по пожарной безопасности и тренировки действий в случае возникновения пожаров.
- Необходимо правильное содержание и эксплуатация электрических приборов.

В случае возникновения пожара необходимо:

- Произвести вызов пожарной службы.
- Предпринять меры по эвакуации персонала из офисного помещения.
- При отсутствии прямых угроз здоровью и жизни произвести попытку тушения возникшего возгорания.

В производственных помещениях, где расположены рабочие места с ВДТ и ПК, применяются главным образом углекислотные огнетушители (ОУ-5), достоинствами которых являются высокая эффективность тушения пожара, сохранность электронного оборудования, диэлектрические свойства углекислого газа, что позволяет использовать эти огнетушители даже в том случае, когда не удастся обесточить электроустановку сразу.

## 5.5. Выводы по разделу

Все производственные факторы на рабочем месте соответствуют нормам, которые были продемонстрированы в данном разделе. Для уменьшения влияния на организм человека физиологического и психофизиологического воздействия, необходимо следовать мерам, указанным в МР 2.2.9.2311-07 [24].

Категория помещения по электробезопасности, согласно ПУЭ, соответствует первому классу – «помещения без повышенной опасности» [28].

Категория тяжести труда в офисе по СанПиН 1.2.3685-21 относится к категории Ia (работы, производимые сидя и сопровождающиеся незначительным физическим напряжением) [20].

Помещение, в котором проводилось исследование, относится к категории помещений группы В1-В4. Характеристика веществ и материалов, находящихся в помещении: горючие и трудногорючие жидкости, твёрдые 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 находятся (обращаются), не относятся к категории А или Б [29].

Согласно постановлению «Об утверждении критериев отнесения объектов, оказывающих негативное воздействие на окружающую среду, к объектам I, II, III и IV категорий», рассмотренный объект оказывает незначительное негативное воздействие на окружающую среду и относится к объектам III категории [30].

## ЗАКЛЮЧЕНИЕ

Таким образом, цель данной работы, а именно, разработка сервиса, способного автоматически выявлять заболевания сельскохозяйственных культур на основе анализа больших данных, была достигнута.

В ходе выполнения работы были решены следующие поставленные задачи:

1. Произведен аналитический обзор существующих исследований на тему, выделены их слабые стороны.
2. Подготовлен набор больших данных для обучения модели нейронной сети.
3. Реализован алгоритм обнаружения болезней сельскохозяйственных культур с использованием архитектуры нейронных сетей EfficientNet.
4. Модель нейронной сети была обучена на наборе больших данных, проведен сравнительный анализ точности обнаружения разработанного алгоритма с существующими.
5. На основе обученной модели был разработан сервис обнаружения болезней, для его реализации использовался язык Python и его библиотека Flask.



## СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Астахов, В. С. Факторы, влияющие на урожайность сельскохозяйственных культур и их качество // В. С. Астахов // Инновационные решения в технологиях и механизации сельскохозяйственного производства: сб. науч. тр. / редкол.: В. Р. Петровец [и др.]. – Горки : БГСХА, 2020. – Вып. 5. – С. 327-332.
2. Tirkey, D., Singh, K. K., & Tripathi, S. (2023). Performance analysis of AI-based solutions for crop disease identification, detection, and classification. B Smart Agricultural Technology (Т. 5, с. 100238). Elsevier BV. <https://doi.org/10.1016/j.atech.2023.100238>
3. Saleem, Potgieter, & Mahmood Arif. (2019). Plant Disease Detection and Classification by Deep Learning. B Plants (Т. 8, Issue 11, с. 468). MDPI AG. <https://doi.org/10.3390/plants8110468>
4. Mohameth, F., Bingcai, C., & Sada, K. A. (2020). Plant Disease Detection with Deep Learning and Feature Extraction Using Plant Village. B Journal of Computer and Communications (Т. 08, Issue 06, сс. 10–22). Scientific Research Publishing, Inc. <https://doi.org/10.4236/jcc.2020.86002>
5. Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using Deep Learning for Image-Based Plant Disease Detection. B Frontiers in Plant Science (Т. 7). Frontiers Media SA. <https://doi.org/10.3389/fpls.2016.01419>
6. Argüeso, D., Picon, A., Irusta, U., Medela, A., San-Emeterio, M. G., Bereciartua, A., & Alvarez-Gila, A. (2020). Few-Shot Learning approach for plant disease classification using images taken in the field. B Computers and Electronics in Agriculture (Т. 175, с. 105542). Elsevier BV. <https://doi.org/10.1016/j.compag.2020.105542>
7. Tan M., Le Q. V. Efficientnet: Rethinking model scaling for convolutional neural networks // arXiv preprint arXiv:1905.11946. - 2019.
8. Noyan, M. A. (2022). Uncovering bias in the PlantVillage dataset. arXiv. <https://doi.org/10.48550/ARXIV.2206.04374>

9. Fabi Nahian Madhurja. (2022, November). Plant Disease [50 Classes], Version 1. Retrieved December 20, 2023 from <https://www.kaggle.com/datasets/fabinahian/plant-disease-50-classes>.
10. SkillFactory. Flask [Электронный ресурс] // SkillFactory. – URL: <https://blog.skillfactory.ru/glossary/flask/> (дата обращения: 14.04.2023).
11. Hostinger. What Is HTML? Hypertext Markup Language Basics Explained [Электронный ресурс] // Hostinger. – URL: <https://www.hostinger.com/tutorials/what-is-html> (дата обращения: 14.04.2023).
12. Educative. What is CSS? [Электронный ресурс] // Educative. – URL: <https://www.educative.io/answers/what-is-css> (дата обращения: 14.04.2023).
13. TensorFlow. Why TensorFlow [Электронный ресурс] // TensorFlow. – URL: <https://www.tensorflow.org/about> (дата обращения: 15.04.2023).
14. Keras. About Keras [Электронный ресурс] // Keras. – URL: <https://keras.io/about/> (дата обращения: 15.04.2023).
15. Приказ № 6-181 от 18.11.2022 О тарифах на электрическую энергию для населения и приравненных к нему категорий потребителей Томской области на 2023 год [Электронный ресурс] // Томскэнергосбыт. 2022. – URL: <https://tomskenergobyt.ru/individuals/tariffes.php> (дата обращения: 16.06.2023).
16. ГОСТ 22269-76 Рабочее место оператора. Взаимное расположение элементов рабочего места.
17. ГОСТ 12.2.032-78 Система стандартов безопасности труда (ССБТ). Рабочее место при выполнении работ сидя. Общие эргономические требования.
18. ГОСТ Р 50923-96 Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения.
19. СП 52.13330.2016 Естественное и искусственное освещение. Актуализированная редакция СНиП 23-05-95.

- 20.СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания.
- 21.ГОСТ 12.1.005-88 Система стандартов безопасности труда (ССБТ). Общие санитарно-гигиенические требования к воздуху рабочей зоны.
- 22.СП 51.13330.2011 Защита от шума. Актуализированная редакция СНиП 23-03-2003.
- 23.ГОСТ 12.1.006-84 Система стандартов безопасности труда (ССБТ). Электромагнитные поля радиочастот. Допустимые уровни на рабочих местах и требования к проведению контроля
- 24.МР 2.2.9.2311-07 Профилактика стрессового состояния работников при различных видах профессиональной деятельности. Методические рекомендации.
- 25.ГОСТ 17.4.3.04-85 Охрана природы. Почвы. Общие требования к контролю и охране от загрязнения.
- 26.ГОСТ Р 53692-2009 Ресурсосбережение. Обращение с отходами. Этапы технологического цикла отходов.
- 27.ГОСТ 12.1.004-91 Система стандартов безопасности труда (ССБТ). Пожарная безопасность. Общие требования.
- 28.Правила устройства электроустановок (ПУЭ). Седьмое издание.
- 29.СП 12.13130.2009 Определение категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности.
- 30.Приказ Об утверждении критериев отнесения объектов, оказывающих негативное воздействие на окружающую среду, к объектам I, II, III и IV категорий от 31.12.2020 года N 2398 (с изм. от 07.10.2021).

# ПРИЛОЖЕНИЕ А

(обязательное)

## Chapter 1 Analysis of the subject area

Студент:

Группа	ФИО	Подпись	Дата
8ВМ1И	Жиганов Владислав Вадимович		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент (ОИТ ИШИТР)	Ковин Р. В.	К.Т.Н., доцент		

Консультант — лингвист

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель (ОИЯ, ШБИП)	Ануфриева Т. Н.			

## **CHAPTER 1. ANALYSIS OF THE SUBJECT AREA**

### **1.1. Description of the subject area**

As the world's population continues to grow rapidly, the challenge of providing enough food becomes more pressing. Farming crops is essential in feeding the people of our planet. Many factors can affect crop yields and these can be categorized into three groups: objective, subjective, and indirect.

Subjective factors are those influenced by human actions. These include the quality, type, and timing of soil preparation before planting; the timing, quality, and method of planting; the seed variety, quality, consistency, and protection against pests and diseases; the timing, amount, and quality of fertilizer used; the amount, quality, and timing of irrigation or drainage; the promptness and methods used to control weeds, pests, and diseases; as well as new farming methods and agronomic techniques [1].

Experts use a variety of techniques to identify plant diseases. These include visual inspection, microscopic analysis of morphological traits to identify pathogens, and the use of molecular, serological, and microbiological diagnostic methods.

Manually identifying plant diseases can be time-consuming and prone to errors, making it an unreliable method for preventing their spread. However, the use of advanced technologies like machine learning (ML) and deep learning (DL) can help overcome these issues by providing early and accurate detection of plant diseases.

### **1.2. Overview of existing solutions**

The growing use of artificial intelligence in detecting plant diseases can be attributed to several factors. One such factor is the ability of AI-driven solutions to facilitate early identification and prevention of crop diseases [2].

Since the introduction of AlexNet, a convolutional neural network architecture for image detection, segmentation and classification, many modern DL models and architectures have evolved. This section discusses research that has been conducted using DL architectures to identify and classify plant diseases [3]. The

algorithms discussed below are compared with the algorithm that has been developed.

### **1.2.1. ResNet-50 with SVM**

Mohameth et al. utilized a variety of contemporary CNN architectures and multiple classifiers to automatically detect plant diseases in the PlantVillage dataset. For feature extraction, they employed VGG-16, ResNet-50, and GoogLeNet CNN architectures. For classification, they used the k-nearest neighbors method and support vector machine (SVM). Their findings revealed that the combination of SVM with ResNet-50 yielded the best results with 98% accuracy [4].

### **1.2.2. GoogLeNet**

Mohanty et al. evaluated the effectiveness of the AlexNet and GoogLeNet CNN architectures in identifying plant diseases using the PlantVillage dataset. They carried out 60 experiments with 60 distinct configurations. Their findings indicated that GoogLeNet, when used with transfer learning, achieved the best results with an accuracy of 99.3%. The article states that GoogLeNet utilized approximately 27 million training parameters [5].

### **1.2.3. Inception v3 transferred to target domain SVM**

Argüeso et al. utilized the Few-shot learning (FSL) algorithm to classify plant diseases. They partitioned the PlantVillage dataset images into source and target domains. In the source domain, they fine-tuned the Inception V3 network to examine general plant leaf characteristics. These characteristics were subsequently transferred to the target domain to investigate new leaf types using a limited number of images. A Siamese neural network with a Triplet loss function was employed for the FSL method. Their findings indicated that the dataset size could be reduced by 89.1% while only sacrificing 4% accuracy, making this approach suitable for small training sets [6].

## **1.3. Conclusion on the analysis of the subject area**

While the research studies mentioned above have been successful in accurately recognizing plant diseases to a certain extent, their approaches are not without their limitations and shortcomings:

1. The PlantVillage dataset is commonly used among researchers. While using a shared dataset can make it easier to compare different models and methods, relying solely on this dataset for training and evaluation can result in overfitting.
2. Due to the vast and unorganized nature of agricultural data, it is necessary to have a big data environment in place to effectively analyze such datasets.
3. The majority of the research studies examined concentrate on particular plant species or specific diseases affecting them.
4. One major limitation of the research studies mentioned earlier is their reliance on an excessive number of training parameters.

Therefore, it is crucial to conduct additional research to create dependable models capable of accurately recognizing diseases in various plant species.

## ПРИЛОЖЕНИЕ Б

### Обучение модели EfficientNetV4 на наборе больших данных

(справочное)

```
import numpy as np
import pandas as pd
import os
import time
import matplotlib.pyplot as plt
import cv2
import seaborn as sns
import shutil
from sklearn.metrics import confusion_matrix, classification_report
from sklearn.model_selection import train_test_split
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.layers import Dense, Activation, Dropout, Conv2D,
MaxPooling2D, BatchNormalization
from tensorflow.keras.optimizers import Adam, Adamax
from tensorflow.keras.metrics import categorical_crossentropy
from tensorflow.keras import regularizers
from tensorflow.keras.models import Model

sdir=r'../input/plant-disease/ '
min_samples=40
filepaths = []
labels=[]
classlist=os.listdir(sdir)
for klass in classlist:
    classpath=os.path.join(sdir, klass)
    flist=os.listdir(classpath)
    if len(flist) >= min_samples:
        for f in flist:
            fpath=os.path.join(classpath,f)
            filepaths.append(fpath)
            labels.append(klass)

Fseries=pd.Series(filepaths, name='filepaths')
Lseries=pd.Series(labels, name='labels')
df=pd.concat([Fseries, Lseries], axis=1)
train_df, dummy_df=train_test_split(df, train_size=.9, shuffle=True,
random_state=123, stratify=df['labels'])
valid_df, test_df=train_test_split(dummy_df, train_size=.5, shuffle=True,
random_state=123, stratify=dummy_df['labels'])
print('train_df length: ', len(train_df), ' test_df length: ',
len(test_df), ' valid_df length: ', len(valid_df))
classes=sorted(list(train_df['labels'].unique()))
class_count = len(classes)
print('The number of classes in the dataset is: ', class_count)
```



```

groups=train_df.groupby('labels')
print('{0:^30s} {1:^13s}'.format('CLASS', 'IMAGE COUNT'))
countlist=[]
classlist=[]
for label in sorted(list(train_df['labels'].unique())):
    group=groups.get_group(label)
    countlist.append(len(group))
    classlist.append(label)
    print('{0:^30s} {1:^13s}'.format(label, str(len(group))))

max_value=np.max(countlist)
max_index=countlist.index(max_value)
max_class=classlist[max_index]
min_value=np.min(countlist)
min_index=countlist.index(min_value)
min_class=classlist[min_index]
print(max_class, ' has the most images= ',max_value, ' ', min_class, '
has the least images= ', min_value)
ht=0
wt=0
train_df_sample=train_df.sample(n=100, random_state=123,axis=0)
for i in range (len(train_df_sample)):
    fpath=train_df_sample['filepaths'].iloc[i]
    img=plt.imread(fpath)
    shape=img.shape
    ht += shape[0]
    wt += shape[1]
print('average height= ', ht//100, ' average width= ', wt//100, ' aspect
ratio= ', ht/wt)

def trim(df, max_samples, min_samples, column):
    df=df.copy()
    groups=df.groupby(column)
    trimmed_df = pd.DataFrame(columns = df.columns)
    groups=df.groupby(column)
    for label in df[column].unique():
        group=groups.get_group(label)
        count=len(group)
        if count > max_samples:
            sampled_group=group.sample(n=max_samples,
random_state=123,axis=0)
            trimmed_df=pd.concat([trimmed_df, sampled_group], axis=0)
        else:
            if count>=min_samples:
                sampled_group=group
                trimmed_df=pd.concat([trimmed_df, sampled_group], axis=0)
    print('after trimming, the maximum samples in any class is now
',max_samples, ' and the minimum samples in any class is ', min_samples)
    return trimmed_df

max_samples=200

```

```

min_samples=36
column='labels'
train_df= trim(train_df, max_samples, min_samples, column)

def balance(df, n, working_dir, img_size):
    df=df.copy()
    print('Initial length of dataframe is ', len(df))
    aug_dir=os.path.join(working_dir, 'aug')
    if os.path.isdir(aug_dir):
        shutil.rmtree(aug_dir)
    os.mkdir(aug_dir)
    for label in df['labels'].unique():
        dir_path=os.path.join(aug_dir,label)
        os.mkdir(dir_path)

    total=0
    gen=ImageDataGenerator(horizontal_flip=True, rotation_range=20,
width_shift_range=.2,
                                height_shift_range=.2, zoom_range=.2)
    groups=df.groupby('labels')
    for label in df['labels'].unique():
        group=groups.get_group(label)
        sample_count=len(group)
        if sample_count< n:
            aug_img_count=0
            delta=n - sample_count
            target_dir=os.path.join(aug_dir, label)
            msg='{0:40s} for class {1:^30s} creating {2:^5s} augmented
images'.format(' ', label, str(delta))
            print(msg, '\r', end='')
            aug_gen=gen.flow_from_dataframe( group, x_col='filepaths',
y_col=None, target_size=img_size,
                                class_mode=None,
batch_size=1, shuffle=False,
                                save_to_dir=target_dir,
save_prefix='aug-', color_mode='rgb',
                                save_format='jpg')

            while aug_img_count<delta:
                images=next(aug_gen)
                aug_img_count += len(images)
            total +=aug_img_count
    print('Total Augmented images created= ', total)
    aug_fpaths=[]
    aug_labels=[]
    classlist=os.listdir(aug_dir)
    for klass in classlist:
        classpath=os.path.join(aug_dir, klass)
        flist=os.listdir(classpath)
        for f in flist:
            fpath=os.path.join(classpath,f)

```

```

        aug_fpaths.append(fpath)
        aug_labels.append(klass)
    Fseries=pd.Series(aug_fpaths, name='filepaths')
    Lseries=pd.Series(aug_labels, name='labels')
    aug_df=pd.concat([Fseries, Lseries], axis=1)
    df=pd.concat([df,aug_df], axis=0).reset_index(drop=True)
    print('Length of augmented dataframe is now ', len(df))
    return df

n=200
working_dir=r'./'
img_size=(200,200)
train_df=balance(train_df, n, working_dir, img_size)

batch_size=20
trgen=ImageDataGenerator(horizontal_flip=True,rotation_range=20,
width_shift_range=.2,
                                height_shift_range=.2, zoom_range=.2 )
t_and_v_gen=ImageDataGenerator()
msg='{0:70s} for train generator'.format(' ')
print(msg, '\r', end='')
train_gen=trgen.flow_from_dataframe(train_df, x_col='filepaths',
y_col='labels', target_size=img_size,
                                class_mode='categorical',
color_mode='rgb', shuffle=True, batch_size=batch_size)
msg='{0:70s} for valid generator'.format(' ')
print(msg, '\r', end='')
valid_gen=t_and_v_gen.flow_from_dataframe(valid_df, x_col='filepaths',
y_col='labels', target_size=img_size,
                                class_mode='categorical',
color_mode='rgb', shuffle=False, batch_size=batch_size)
length=len(test_df)
test_batch_size=sorted([int(length/n) for n in range(1,length+1) if
length % n ==0 and length/n<=80],reverse=True)[0]
test_steps=int(length/test_batch_size)
msg='{0:70s} for test generator'.format(' ')
print(msg, '\r', end='')
test_gen=t_and_v_gen.flow_from_dataframe(test_df, x_col='filepaths',
y_col='labels', target_size=img_size,
                                class_mode='categorical',
color_mode='rgb', shuffle=False, batch_size=test_batch_size)

classes=list(train_gen.class_indices.keys())
class_indices=list(train_gen.class_indices.values())
class_count=len(classes)
labels=test_gen.labels
print ( 'test batch size: ' ,test_batch_size, ' test steps: ',
test_steps, ' number of classes : ', class_count)

def show_image_samples(gen ):
    t_dict=gen.class_indices

```

```

classes=list(t_dict.keys())
images,labels=next(gen)
plt.figure(figsize=(20, 20))
length=len(labels)
if length<25:
    r=length
else:
    r=25
for i in range(r):
    plt.subplot(5, 5, i + 1)
    image=images[i] /255
    plt.imshow(image)
    index=np.argmax(labels[i])
    class_name=classes[index]
    plt.title(class_name, color='blue', fontsize=14)
    plt.axis('off')
plt.show()

```

```
show_image_samples(train_gen )
```

```

img_shape=(img_size[0], img_size[1], 3)
model_name='EfficientNetB4'
base_model=tf.keras.applications.efficientnet.EfficientNetB4(include_top=
False, weights="imagenet",input_shape=img_shape, pooling='max')
base_model.trainable=True
x=base_model.output
x=BatchNormalization(axis=-1, momentum=0.99, epsilon=0.001 )(x)
x = Dense(256, kernel_regularizer = regularizers.l2(1 =
0.016),activity_regularizer=regularizers.l1(0.006),
        bias_regularizer=regularizers.l1(0.006)
,activation='relu')(x)
x=Dropout(rate=.4, seed=123)(x)
output=Dense(class_count, activation='softmax')(x)
model=Model(inputs=base_model.input, outputs=output)
lr=.001
model.compile(Adamax(learning_rate=lr), loss='categorical_crossentropy',
metrics=['accuracy'])

```

```

class LR_ASK(keras.callbacks.Callback):
    def __init__(self, model, epochs, ask_epoch):
        super(LR_ASK, self).__init__()
        self.model=model
        self.ask_epoch=ask_epoch
        self.epochs=epochs
        self.ask=True
        self.lowest_vloss=np.inf
        self.best_weights=self.model.get_weights()
self.best_epoch=1

```

```

def on_train_begin(self, logs=None):
    if self.ask_epoch == 0:
        print('you set ask_epoch = 0, ask_epoch will be set to 1',
flush=True)
        self.ask_epoch=1
    if self.ask_epoch >= self.epochs:
        print('ask_epoch >= epochs, will train for ', epochs, '
epochs', flush=True)
        self.ask=False
    if self.epochs == 1:
        self.ask=False
    else:
        print('Training will proceed until epoch', ask_epoch,' then
you will be asked to')
        print(' enter H to halt training or enter an integer for how
many more epochs to run then be asked again')
        self.start_time= time.time()

def on_train_end(self, logs=None):
    print('loading model with weights from epoch ', self.best_epoch)
    self.model.set_weights(self.best_weights)
    tr_duration=time.time() - self.start_time
    hours = tr_duration // 3600
    minutes = (tr_duration - (hours * 3600)) // 60
    seconds = tr_duration - ((hours * 3600) + (minutes * 60))
    msg = f'training elapsed time was {str(hours)} hours,
{minutes:4.1f} minutes, {seconds:4.2f} seconds)'
    print (msg, flush=True)

def on_epoch_end(self, epoch, logs=None):
    v_loss=logs.get('val_loss')
    if v_loss< self.lowest_vloss:
        self.lowest_vloss=v_loss
        self.best_weights=self.model.get_weights()
        self.best_epoch=epoch + 1
        print (f'\n validation loss of {v_loss:7.4f} is below lowest
loss, saving weights from epoch {str(epoch + 1):3s} as best weights')
    else:
        print (f'\n validation loss of {v_loss:7.4f} is above lowest
loss of {self.lowest_vloss:7.4f} keeping weights from epoch
{str(self.best_epoch)} as best weights')

    if self.ask:
        if epoch + 1 ==self.ask_epoch:
            print('\n Enter H to end training or an integer for the
number of additional epochs to run then ask again')
            ans=input()

            if ans == 'H' or ans =='h' or ans == '0':
                print ('you entered ', ans, ' Training halted on
epoch ', epoch+1, ' due to user input\n', flush=True)

```

```

        self.model.stop_training = True
    else:
        self.ask_epoch += int(ans)
        if self.ask_epoch > self.epochs:
            print('\nYou specified maximum epochs of as ',
self.epochs, ' cannot train for ', self.ask_epoch, flush=True)
        else:
            print ('you entered ', ans, ' Training will
continue to epoch ', self.ask_epoch, flush=True)

lr=float(tf.keras.backend.get_value(self.model.optimizer.lr))
        print(f'current LR is {lr:7.5f} hit enter to
keep this LR or enter a new LR')
        ans=input(' ')
        if ans == '':
            print (f'keeping current LR of {lr:7.5f}')
        else:
            new_lr=float(ans)

tf.keras.backend.set_value(self.model.optimizer.lr, new_lr)
        print(' changing LR to ', ans)

epochs=40
ask_epoch=5
ask=LR_ASK(model, epochs, ask_epoch)
callbacks=[ask]

history=model.fit(x=train_gen, epochs=epochs, verbose=1,
callbacks=callbacks, validation_data=valid_gen,
validation_steps=None, shuffle=False, initial_epoch=0)

def tr_plot(tr_data, start_epoch):
    tacc=tr_data.history['accuracy']
    tloss=tr_data.history['loss']
    vacc=tr_data.history['val_accuracy']
    vloss=tr_data.history['val_loss']
    Epoch_count=len(tacc)+ start_epoch
    Epochs=[]
    for i in range (start_epoch ,Epoch_count):
        Epochs.append(i+1)
    index_loss=np.argmin(vloss)
    val_lowest=vloss[index_loss]
    index_acc=np.argmax(vacc)
    acc_highest=vacc[index_acc]
    plt.style.use('fivethirtyeight')
    sc_label='best epoch= '+ str(index_loss+1 +start_epoch)
    vc_label='best epoch= '+ str(index_acc + 1+ start_epoch)
    fig,axes=plt.subplots(nrows=1, ncols=2, figsize=(20,8))
    axes[0].plot(EPOCHS,tloss, 'r', label='Training loss')

```

```

    axes[0].plot(Epochs,vloss,'g',label='Validation loss' )
    axes[0].scatter(index_loss+1 +start_epoch,val_lowest, s=150, c=
'blue', label=sc_label)
    axes[0].set_title('Training and Validation Loss')
    axes[0].set_xlabel('Epochs')
    axes[0].set_ylabel('Loss')
    axes[0].legend()
    axes[1].plot (Epochs,tacc,'r',label= 'Training Accuracy')
    axes[1].plot (Epochs,vacc,'g',label= 'Validation Accuracy')
    axes[1].scatter(index_acc+1 +start_epoch,acc_highest, s=150, c=
'blue', label=vc_label)
    axes[1].set_title('Training and Validation Accuracy')
    axes[1].set_xlabel('Epochs')
    axes[1].set_ylabel('Accuracy')
    axes[1].legend()
    plt.tight_layout
    plt.show()

```

```
tr_plot(history,0)
```

```

def predictor(test_gen, test_steps):
    y_pred= []
    y_true=test_gen.labels
    classes=list(test_gen.class_indices.keys())
    class_count=len(classes)
    errors=0
    preds=model.predict(test_gen, verbose=1)
    tests=len(preds)
    for i, p in enumerate(preds):
        pred_index=np.argmax(p)
        true_index=test_gen.labels[i]
        if pred_index != true_index:
            y_pred.append(pred_index)

    acc=( 1-errors/tests) * 100
    print(f'there were {errors} errors in {tests} tests for an accuracy
of {acc:6.2f}')
    ypred=np.array(y_pred)
    ytrue=np.array(y_true)
    if class_count <=30:
        cm = confusion_matrix(ytrue, ypred )
        plt.figure(figsize=(12, 8))
        sns.heatmap(cm, annot=True, vmin=0, fmt='g', cmap='Blues',
cbar=False)
        plt.xticks(np.arange(class_count)+.5, classes, rotation=90)
        plt.yticks(np.arange(class_count)+.5, classes, rotation=0)
        plt.xlabel("Predicted")
        plt.ylabel("Actual")
        plt.title("Confusion Matrix")
        plt.show()

```

```
    clr = classification_report(y_true, y_pred, target_names=classes,
digits= 4)
    print("Classification Report:\n-----\n", clr)
    return errors, tests
errors, tests=predictor(test_gen, test_steps)

subject='plant disease'
acc=str(( 1-errors/tests) * 100)
index=acc.rfind('.')
acc=acc[:index + 3]
save_id= subject + '_' + str(acc) + '.h5'
model_save_loc=os.path.join(working_dir, save_id)
model.save(model_save_loc)
print ('model was saved as ' , model_save_loc )
```