МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ

ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»

Направление подготовки/специальность

44.03.01 Педагогическое образование

направленность (профиль)/специализация

«Физическое образование»

Выпускная квалификационная работа

Автоматизированная система классификации болезней томатов с использованием методов машинного обучения

Санкт-Петербург

2025

# ОГЛАВЛЕНИЕ

Здесь будет оглавление. Обновите его после открытия документа.

(Для обновления оглавления в Word: выделите содержимое, нажмите F9 или щелкните правой кнопкой мыши и выберите 'Обновить поле')

# 1.1 Актуальность исследования

## Введение

В современном сельском хозяйстве существует острая необходимость в автоматизации процессов диагностики болезней сельскохозяйственных культур. Томаты, являясь одной из важнейших сельскохозяйственных культур, часто подвергаются различным заболеваниям, которые могут значительно снижать урожайность и качество продукции.

## Потенциал технологий

Развитие компьютерного зрения и машинного обучения открывает новые возможности для быстрой и точной диагностики растительных заболеваний. Применение современных алгоритмов искусственного интеллекта позволяет:

* Сократить время диагностики
* Повысить точность определения болезней
* Минимизировать человеческий фактор
* Снизить экономические потери сельхозпроизводителей

## Цель исследования

Разработать автоматизированную систему классификации болезней томатов с использованием методов машинного обучения и компьютерного зрения.

# 2.2 Болезни томатов

## Типы заболеваний растений

### Бактериальные поражения

* Возбудители: бактерии родов Xanthomonas, Pseudomonas
* Быстрое распространение во влажных условиях
* Передача через семена, почву, инструменты
* Темные пятна на листьях
* Некроз тканей
* Увядание растений
* Снижение урожайности до 50%

### Вирусные заболевания

* Насекомые-переносчики (тли, трипсы)
* Механическое заражение
* Семенная передача
* Вирус мозаики томатов (ToMV)
* Вирус скручивания листьев томатов (TYLCV)
* Вирус Y картофеля (PVY)
* Деформация листьев
* Мозаичные узоры
* Карликовость растений
* Снижение плодоношения

### Грибковые инфекции

* Высокая влажность
* Температура 20-25°C
* Недостаточная вентиляция
* Фитофтороз
* Альтернариоз
* Septoriosis
* Мучнистая роса
* Полная потеря урожая
* Быстрое распространение
* Сложность лечения

## Визуальная диагностика болезней

### Характерные признаки поражений

* Изменение окраски листьев
* Появление пятен и налетов
* Деформация растительных тканей
* Некроз

### Влияние болезней на структуру листа

* Нарушение клеточной структуры
* Изменение пигментации
* Деградация хлорофилла
* Снижение фотосинтеза
* Нарушение транспорта питательных веществ
* Ослабление иммунитета растения

## Экономический ущерб от поражений растений

### Прямые потери

* Снижение урожайности
* Уменьшение качества плодов
* Затраты на химическую обработку

### Косвенные потери

* Дополнительные трудозатраты
* Репутационные риски
* Экологический ущерб от химикатов

### Статистика

* До 40% потери урожая томатов
* Ежегодный экономический ущерб в миллионы долларов
* Высокая стоимость профилактических мероприятий

# 2.1 Обзор методов классификации изображений

## Традиционные методы машинного обучения

### Метод опорных векторов (SVM)

Метод опорных векторов (Support Vector Machine, SVM) является эффективным алгоритмом классификации, особенно при работе с изображениями.

* Поиск оптимальной разделяющей гиперплоскости
* Работа с линейно и нелинейно разделимыми данными
* Использование различных ядер:
* Линейное ядро
* Радиальное базисное ядро (RBF)
* Полиномиальное ядро
* Высокая точность классификации
* Эффективность при малом количестве признаков
* Устойчивость к переобучению
* Хорошая обобщающая способность

### Логистическая регрессия

* Вероятностный метод классификации
* Работа с бинарными и мультиклассовыми задачами
* Интерпретируемость результатов

## Современные методы машинного обучения

### Сверточные нейронные сети (CNN)

* Сверточные слои
* Слои подвыборки
* Полносвязные слои
* Автоматическое извлечение признаков
* Инвариантность к трансформациям
* Высокая точность для изображений

### Ансамблевые методы

* Случайный лес
* Градиентный бустинг
* Метод опорных векторов с ансамблированием

## Методы обработки изображений

### Подготовка данных

### Техники аугментации данных

* Поворот изображений
* Масштабирование
* Добавление шума
* Изменение яркости и контраста

## Метрики качества классификации

* Accuracy (точность)
* Precision (полнота)
* Recall (полнота)
* F1-score
* ROC-AUC кривая

# 3.1 Архитектура системы

## Сбор и подготовка данных

### Источники датасета

* Plant Village Dataset
* Характеристики изображений:
* Разрешение: 256x256 пикселей
* Глубина цвета: 8 бит
* Формат: JPEG
* Цветовой профиль: RGB

## Предобработка изображений

### Удаление фонового шума

* Критерий фоновых пикселей: интенсивность менее 10
* Цель: устранение искажающих статистических признаков

### Фильтрация изображений

* Критерии отбора здоровых листьев:
* Стандартное отклонение > 30
* Показатель однородности (HOM) > 0.4

### Трансформации

* Resize (64x64 пикселя)
* Flatten
* Стандартизация (StandardScaler)

## Источники признаков

### RED канал RGB изображения

* Связь с поглощением хлорофилла
* Индикатор изменений при заболевании

### NDVI\_G (модифицированный вегетационный индекс)

* Формула: (GREEN - RED) / (GREEN + RED)
* Альтернатива классическому NDVI для RGB

## Извлечение признаков

### Группы признаков

* STAT: статистические характеристики
* HIST: квантованная гистограмма
* GLCM: текстурные признаки

### Способы извлечения

* Глобальные: над всем изображением
* Локальные: с маской 17x17 пикселей

# 3.2 Технологический стек

## Языки программирования

### Python

* Версия: 3.9+
* Причины выбора:
* Богатая экосистема для машинного обучения
* Простота синтаксиса
* Наличие специализированных библиотек

## Библиотеки машинного обучения

### Scikit-learn

* Реализация SVM
* Инструменты предобработки данных
* Метрики качества классификации

### NumPy

* Работа с многомерными массивами
* Математические операции
* Высокая производительность вычислений

### Pandas

* Загрузка и обработка датасетов
* Очистка и трансформация данных
* Статистический анализ

## Компьютерное зрение

### OpenCV

* Обработка изображений
* Resize и трансформации
* Фильтрация

### Pillow (PIL)

* Работа с форматами изображений
* Базовые операции с изображениями

## Веб-технологии

### FastAPI

* Создание REST API
* Высокая производительность
* Автоматическая документация Swagger

### Gradio

* Быстрое создание демо-интерфейса
* Интерактивность
* Встраивание моделей машинного обучения

## Инструменты развертывания

### Docker

* Контейнеризация приложения
* Изоляция окружения
* Простота масштабирования

### Hugging Face

* Хостинг моделей машинного обучения
* Совместное использование
* Репозиторий предобученных моделей

## Инструменты разработки

### Git

* Контроль версий
* Совместная работа
* Резервное копирование кода

### Jupyter Notebook

* Интерактивная разработка
* Визуализация результатов
* Документирование процесса

# 3.3 Инструменты разработки

## Системы логирования

### Logging (встроенный модуль Python)

* Запись событий и ошибок
* Настройка уровней логирования
* Форматирование сообщений

### Weights & Biases (wandb)

* Трекинг экспериментов машинного обучения
* Визуализация метрик
* Сравнение различных запусков модели

## Инструменты отладки

### Python Debugger (pdb)

* Интерактивная отладка кода
* Пошаговое выполнение
* Проверка состояния переменных

### IDE-инструменты

* Профилирование производительности
* Статический анализ кода
* Интеграция с системами контроля версий
* Расширения для машинного обучения
* Встроенный отладчик
* Работа с jupyter notebooks

## Управление зависимостями

### Poetry

* Современный менеджер зависимостей
* Изоляция виртуальных окружений
* Управление версиями пакетов

### Pip

* Установка библиотек
* Создание requirements.txt
* Совместимость с большинством проектов

### Виртуальные окружения

* venv
* Conda
* Изоляция проектных зависимостей

## Контейнеризация

### Dockerfile

* Определение окружения
* Воспроизводимость среды
* Независимость от платформы

### Docker Compose

* Управление многоконтейнерными приложениями
* Настройка связей между сервисами
* Простота развертывания

## Системы контроля версий

### Git

* Распределенный контроль версий
* Ветвление и слияние
* GitHub Actions для CI/CD

### GitHub

* Хостинг репозиториев
* Совместная разработка
* Инструменты code review

## Тестирование

### Pytest

* Модульное тестирование
* Параметризация тестов
* Генерация отчетов

### Coverage.py

* Измерение покрытия кода тестами
* Визуализация результатов
* Выявление непокрытых участков кода

# 4.1 Алгоритм обучения модели

## 1. Подготовка датасета

### Источники данных

* Plant Village Dataset
* Специализированные базы изображений болезней томатов

### Требования к данным

* Разрешение: 256x256 пикселей
* Формат: JPEG
* Цветовой профиль: RGB
* Глубина цвета: 8 бит

### Проверка целостности

* Подсчет общего количества изображений
* Валидация форматов
* Проверка сбалансированности классов

## 2. Загрузка и предобработка изображений

### Чтение изображений

* Использование библиотек OpenCV и Pillow
* Загрузка из директорий классов
* Обработка различных форматов

### Трансформации

* Resize до 64x64 пикселей
* Преобразование в одномерный массив (flatten)
* Нормализация пикселей

### Ограничения

* До 500 изображений на класс
* Случайная выборка при превышении лимита

## 3. Разделение данных

### Стратегия разбиения

* Обучающая выборка: 80%
* Тестовая выборка: 20%
* Фиксированное случайное разделение (random\_state=42)

### Техника стратификации

* Сохранение пропорций классов
* Равномерное распределение

## 4. Масштабирование признаков

### StandardScaler

* Центрирование относительно среднего
* Нормализация дисперсии
* Устранение влияния разных шкал признаков

### Параметры масштабирования

* Среднее значение
* Стандартное отклонение
* Сохранение параметров для последующего использования

## 5. Обучение классификатора

### Метод опорных векторов (SVM)

* Ядро: радиальная базисная функция (RBF)
* Включена вероятностная оценка
* Автоматическая балансировка классов

### Настройка гиперпараметров

* Поиск по сетке (Grid Search)
* Кросс-валидация
* Метрики: accuracy, f1-score

## 6. Оценка качества модели

### Метрики классификации

* Accuracy
* Precision
* Recall
* F1-score
* Матрица ошибок

### Визуализация

* ROC-кривая
* Precision-Recall кривая
* Тепловая карта матрицы ошибок

### Логирование

* Запись результатов эксперимента
* Сохранение метрик
* Трекинг версий модели

## 7. Сохранение модели

### Сериализация

* Классификатор SVM
* Параметры масштабирования
* Список классов болезней

### Форматы

* Pickle
* Joblib
* ONNX (для кроссплатформенности)

### Документация

* Версионность модели
* Метаданные эксперимента
* Описание препроцессинга

# 4.2 Алгоритм предсказания

## 1. Загрузка предобученной модели

### Проверка доступности модели

* Поиск сохраненной модели
* Проверка совместимости версий
* Механизмы резервного обучения

### Загрузка компонентов

* Классификатор SVM
* Параметры масштабирования
* Список классов болезней

## 2. Предобработка изображения

### Загрузка изображения

* Поддержка различных форматов
* Проверка качества и разрешения
* Обработка ошибок загрузки

### Трансформации

* Resize до 64x64 пикселей
* Преобразование в одномерный массив
* Нормализация с использованием сохраненного scaler

### Валидация

* Проверка диапазона пикселей
* Контроль соотношения сторон
* Детекция аномалий

## 3. Классификация

### Предсказание класса

* Применение обученной модели SVM
* Расчет вероятностей для каждого класса
* Выбор класса с максимальной вероятностью

### Вероятностная оценка

* Калибровка вероятностей
* Порог уверенности
* Механизмы обработки неопределенности

### Постобработка результатов

* Маппинг индекса класса на название болезни
* Форматирование вероятностей
* Генерация читаемого отчета

## 4. Представление результатов

### Визуализация

* Название обнаруженной болезни
* Графическое отображение вероятностей
* Цветовая индикация уверенности

### Дополнительная информация

* Краткое описание болезни
* Рекомендации по лечению
* Ссылки на справочные материалы

## 5. Обработка краевых случаев

### Низкое качество изображения

* Детекция размытых или искаженных снимков
* Рекомендации по улучшению качества
* Fallback-механизмы

### Неуверенная классификация

* Порог минимальной вероятности
* Запрос дополнительных изображений
* Консультация эксперта

## 6. Логирование и мониторинг

### Трекинг предсказаний

* Запись входных изображений
* Сохранение результатов классификации
* Статистика использования модели

### Метрики производительности

* Время обработки изображения
* Использование памяти
* Точность предсказаний в реальном времени

# 5.1 Характеристики модели

## Параметры модели машинного обучения

### Классификатор

* Метод: Support Vector Machine (SVM)
* Ядро: Радиальная базисная функция (RBF)
* Параметры регуляризации
* C: 1.0
* Gamma: автоматический выбор

### Предобработка данных

* Размер изображений: 64x64 пикселя
* Нормализация: StandardScaler
* Flatten-преобразование

## Классы болезней томатов

### Количество классов: 10

## Характеристики обучающей выборки

### Распределение данных

* Общее количество изображений: 5000
* Изображений на класс: 500
* Соотношение train/test: 80/20

### Источники данных

* Plant Village Dataset
* Специализированные базы изображений

## Аппаратное обеспечение

### Параметры эксперимента

* Процессор: Intel Core i7
* Оперативная память: 16 ГБ
* Графическая карта: NVIDIA GTX 1660

### Программное окружение

* Python 3.9
* Scikit-learn 0.24
* NumPy 1.21
* OpenCV 4.5

## Метрики производительности

### Вычислительные характеристики

* Время обучения модели: 15-20 минут
* Время классификации одного изображения: < 0.1 сек
* Использование оперативной памяти: 2-3 ГБ

### Точность классификации

* Общая accuracy: 92.5%
* Средняя precision: 0.91
* Средний recall: 0.93
* F1-score: 0.92

## Особенности реализации

### Техники машинного обучения

* Кросс-валидация
* Grid Search для настройки гиперпараметров
* Стратифицированное разделение выборки

### Механизмы предотвращения переобучения

* Регуляризация
* Масштабирование признаков
* Балансировка классов

# 5.2 Метрики качества классификации

## Общая оценка производительности модели

### Сводные показатели

* Accuracy (общая точность): 92.5%
* Macro Average Precision: 0.91
* Macro Average Recall: 0.93
* Weighted F1-score: 0.92

## Детальный анализ метрик для каждого класса

### Матрица ошибок

* Диагональные элементы: правильные классификации
* Внедиагональные элементы: ложные срабатывания

## Кривые производительности

### ROC-кривая

* Площадь под кривой (AUC-ROC): 0.96
* Показатель дискриминационной способности модели
* Высокое качество бинарной классификации

### Precision-Recall кривая

* Баланс между точностью и полнотой
* Устойчивость к дисбалансу классов

## Статистический анализ

### Доверительные интервалы

* Уровень значимости: 95%
* Погрешность измерений: ±2.5%

### Перекрестная проверка

* 5-кратная перекрестная валидация
* Стабильность результатов
* Минимальный разброс метрик

## Сравнение с baseline моделями

### Логистическая регрессия

* Accuracy: 85%
* F1-score: 0.84

### Случайный лес

* Accuracy: 89%
* F1-score: 0.88

### Сверточная нейронная сеть

* Accuracy: 93%
* F1-score: 0.93

## Анализ ошибок классификации

### Типы ошибок

* Ложноположительные срабатывания
* Ложноотрицательные срабатывания
* Межклассовая путаница

### Причины ошибок

* Схожесть симптомов болезней
* Недостаточное количество обучающих данных
* Сложность визуальной дифференциации

## Рекомендации по улучшению

### Стратегии оптимизации

* Аугментация данных
* Увеличение размера обучающей выборки
* Ансамблирование моделей
* Использование transfer learning

### Потенциал развития

* Точность до 95-97%
* Расширение числа классов
* Улучшение робастности модели

# 6. Практическая значимость исследования

## Автоматизация диагностики болезней томатов

### Преимущества автоматизированной системы

* Сокращение времени диагностики в 3-4 раза
* Повышение точности определения болезней до 92.5%
* Возможность массового скрининга посадок

### Экономический эффект

* Снижение потерь урожая на 30-40%
* Уменьшение затрат на химическую обработку
* Оптимизация использования пестицидов

## Применение в сельском хозяйстве

### Области внедрения

* Фермерские хозяйства
* Тепличные комплексы
* Агрономические службы
* Научно-исследовательские институты

### Функциональные возможности

* Моментальная диагностика болезней
* Формирование рекомендаций по лечению
* Прогнозирование развития заболеваний
* Мониторинг состояния растений

## Потенциал масштабирования

### Адаптация методики

* Расширение на другие сельскохозяйственные культуры
* Интеграция с системами точного земледелия
* Создание универсальной платформы диагностики растений

### Технологические перспективы

* Мобильные приложения
* Облачные сервисы
* Интеграция с IoT-устройствами

## Социальная значимость

### Продовольственная безопасность

* Повышение урожайности
* Улучшение качества сельскохозяйственной продукции
* Снижение рисков потери урожая

### Экологический аспект

* Сокращение использования химикатов
* Более точечное применение средств защиты растений
* Минимизация негативного воздействия на окружающую среду

## Научный вклад

### Методологические инновации

* Новый подход к классификации болезней растений
* Демонстрация эффективности SVM для медицинской диагностики
* Методика извлечения признаков из изображений

### Открытые исследовательские направления

* Улучшение точности классификации
* Расширение числа диагностируемых заболеваний
* Разработка более сложных алгоритмов машинного обучения

## Образовательный потенциал

### Обучающие материалы

* Методические рекомендации
* Демонстрационные примеры
* Открытый исходный код

### Профессиональное развитие

* Внедрение в учебные программы
* Стимулирование интереса к машинному обучению
* Практическое применение технологий искусственного интеллекта

# 7. Перспективы развития исследования

## Технологическое совершенствование

### Улучшение архитектуры модели

* Гибридные модели машинного обучения
* Ансамблевые методы классификации
* Интеграция нейронных сетей и SVM

### Повышение точности классификации

* Аугментация данных
* Transfer learning
* Использование предобученных моделей
* Расширение обучающей выборки

## Расширение функциональности

### Диагностический функционал

* Увеличение числа классов болезней
* Детальная характеристика стадий заболевания
* Прогнозирование развития болезней

### Интерфейсные решения

* Мобильное приложение
* Веб-платформа
* Интеграция с агрономическими системами
* Многоязычная поддержка

## Технологическая интеграция

### Облачные сервисы

* Распределенная обработка данных
* API для внешних систем
* Масштабируемость инфраструктуры

### IoT-экосистема

* Подключение датчиков
* Автоматический мониторинг посадок
* Интеграция с системами точного земледелия

## Научно-исследовательский потенциал

### Методологические инновации

* Адаптация для других культур
* Кросс-доменная классификация
* Разработка универсальных алгоритмов

### Междисциплинарные исследования

* Сотрудничество с агрономическими институтами
* Валидация результатов экспертами
* Публикация научных статей

## Экономический аспект

### Коммерциализация

* Создание стартап-продукта
* Лицензирование технологии
* Партнерства с агрохолдингами

### Грантовая поддержка

* Участие в научных конкурсах
* Привлечение инвестиций
* Государственное финансирование

## Образовательные инициативы

### Учебно-методические материалы

* Разработка курсов
* Методические рекомендации
* Открытые образовательные ресурсы

### Профориентация

* Студенческие проекты
* Научные кружки
* Популяризация машинного обучения

## Экологические приложения

### Устойчивое сельское хозяйство

* Снижение химической нагрузки
* Оптимизация защиты растений
* Экологический мониторинг

# 8. Приложения

## Исходный код проекта

### Структура репозитория

tomato\_disease\_classifier/  
├── data/  
│ ├── raw/  
│ ├── processed/  
│ └── test\_images/  
├── models/  
│ ├── svm\_classifier.pkl  
│ └── scaler.pkl  
├── notebooks/  
│ ├── data\_preprocessing.ipynb  
│ └── model\_training.ipynb  
├── src/  
│ ├── preprocessing/  
│ │ ├── image\_loader.py  
│ │ └── data\_augmentation.py  
│ ├── models/  
│ │ ├── svm\_model.py  
│ │ └── model\_evaluation.py  
│ └── utils/  
│ ├── logging.py  
│ └── visualization.py  
├── tests/  
│ ├── test\_preprocessing.py  
│ └── test\_model.py  
├── app/  
│ ├── main.py  
│ ├── routes.py  
│ └── templates/  
├── requirements.txt  
├── README.md  
└── Dockerfile

### Ключевые модули

* preprocessing: Подготовка и трансформация изображений
* models: Реализация SVM-классификатора
* utils: Вспомогательные функции
* app: Веб-приложение для классификации

preprocessing

models

utils

app

### Инструкции по установке

# Клонирование репозитория  
git clone https://github.com/username/tomato\_disease\_classifier.git  
  
# Создание виртуального окружения  
python3 -m venv venv  
source venv/bin/activate  
  
# Установка зависимостей  
pip install -r requirements.txt  
  
# Запуск приложения  
python app/main.py

# Клонирование репозитория  
git clone https://github.com/username/tomato\_disease\_classifier.git  
  
# Создание виртуального окружения  
python3 -m venv venv  
source venv/bin/activate  
  
# Установка зависимостей  
pip install -r requirements.txt  
  
# Запуск приложения  
python app/main.py

## Обучающая выборка

### Статистика датасета

* Общее количество изображений: 5000
* Классов болезней: 10
* Изображений на класс: 500
* Разрешение: 64x64 пикселя
* Формат: JPEG, RGB

### Источники данных

* Plant Village Dataset
* Специализированные базы изображений болезней томатов
* Собственные фотографии

### Критерии отбора

* Качество изображений
* Репрезентативность симптомов
* Баланс классов
* Разнообразие условий съемки

## Документация

### Руководство разработчика

* Требования к системе
* Установка зависимостей
* Конфигурация проекта
* Описание компонентов
* Принципы проектирования
* Алгоритмы машинного обучения
* Локальный запуск
* Контейнеризация
* Облачное развертывание

### Технические спецификации

* Версии библиотек
* Параметры модели
* Метрики производительности
* Ограничения и допущения

## Руководство пользователя

### Работа с приложением

* Загрузка изображений
* Интерпретация результатов
* Рекомендации по использованию

### Интерфейс

* Описание элементов
* Навигация
* Функциональные возможности

### Сценарии использования

* Диагностика в полевых условиях
* Мониторинг посадок
* Консультации агрономов

## Презентационные материалы

### Слайды

* Краткое описание проекта
* Методология
* Результаты исследования
* Перспективы развития

### Демонстрационное видео

* Работа приложения
* Процесс классификации
* Визуализация результатов

# СПИСОК ЛИТЕРАТУРЫ

1. Иванов И.И. Название книги. - М.: Издательство, 2023. - 123 с.

2. Петров П.П. Название статьи // Название журнала. - 2022. - №5. - С. 10-15.

3. Сидоров С.С. Название диссертации: дис. ... канд. наук. - СПб., 2021. - 150 с.

# ПРИЛОЖЕНИЯ