МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ ТЮМЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Школа компьютерных наук

Кафедра программного обеспечения

Отчет по дисциплине:

“Интеллектуальные системы поддержки принятия решений”

на тему:

Система распознавания и отслеживания образцов керна с использованием технологий компьютерного зрения и анализа изображений для контроля качества в процессе лабораторных исследований на предприятии Новатэк НТЦ

Выполнили

студенты группы:

МОиАИС 21.03

Прощенко Алексей

Фридрих Александр

Научный руководитель: Шенгелия Давид Юзаевич

Тюмень, 2024

ОГЛАВЛЕНИЕ

[ВВЕДЕНИЕ 2](#_Toc185588342)

[ГЛАВА 1. ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ 6](#_Toc185588343)

[1.1 ВВЕДЕНИЕ В ПРЕДМЕТНУЮ ОБЛАСТЬ 6](#_Toc185588344)

[1.2 АНАЛИЗ СУЩЕСТВУЮЩИХ ПОДХОДОВ 7](#_Toc185588345)

[1.3 ИДЕЯ РЕШЕНИЯ 10](#_Toc185588346)

[1.4. ТРЕБОВАНИЯ И ОГРАНИЧЕНИЯ СИСТЕМЫ 11](#_Toc185588347)

[1.5 ТЕОРЕТИЧЕСКОЕ ОБОСНОВАНИЕ ТЕХНОЛОГИЙ 13](#_Toc185588348)

[ГЛАВА 2. РЕАЛИЗАЦИЯ СИСТЕМЫ 16](#_Toc185588349)

[2.1 ПОСТАНОВКА ЗАДАЧИ 16](#_Toc185588350)

[2.2 ОПИСАНИЕ ИСПОЛЬЗУЕМЫХ ТЕХНОЛОГИЙ 17](#_Toc185588351)

[2.3 ОПИСАНИЕ ФАЙЛОВОГО ХРАНИЛИЩА 20](#_Toc185588352)

[2.4 РАЗРАБОТКА НЕЙРОННЫХ СЕТЕЙ 22](#_Toc185588353)

[2.4.1 МОДЕЛЬ YOLO. ДЕТЕКЦИЯ ОБРАЗЦОВ КЕРНА 22](#_Toc185588354)

[2.4.2 МОДЕЛЬ YOLO. ДЕТЕКЦИЯ ТЕКСТА 23](#_Toc185588355)

[2.4.3 МОДЕЛЬ EASYOCR. РАСПОЗНАВАНИЕ ТЕКСТА 25](#_Toc185588356)

[2.5 РАЗРАБОТКА ПРИЛОЖЕНИЯ 25](#_Toc185588357)

[2.5.1 ХОД РАБОТЫ 25](#_Toc185588358)

[2.5.2 ФУНКЦИОНАЛЬНАЯ СХЕМА 26](#_Toc185588359)

[2.5.3 АРХИТЕКТУРА СИСТЕМЫ 28](#_Toc185588360)

[2.5.4 ОПИСАНИЕ МОДУЛЕЙ 30](#_Toc185588361)

[2.5.5 ОПИСАНИЕ ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ 31](#_Toc185588362)

[2.6 ОРГАНИЗАЦИЯ ТЕСТИРОВАНИЯ 33](#_Toc185588363)

[ЗАКЛЮЧЕНИЕ 36](#_Toc185588364)

[СПИСОК ЛИТЕРАТУРЫ 38](#_Toc185588365)

[ПРИЛОЖЕНИЕ 40](#_Toc185588366)

# ВВЕДЕНИЕ

Процесс обработки и исследования образцов керна в лабораторных условиях играет ключевую роль в геологических изысканиях. Образцы керна — цилиндрические вырезки горной породы, извлеченные из скважин — используются для анализа структуры и состава породы, что имеет решающее значение для оценки потенциала месторождений. Однако отсутствие систематизированного отслеживания образцов и управления их перемещением между лабораториями создаёт значительные трудности, приводя к неэффективности и затруднениям в управлении данными.

На каждом этапе лабораторных исследований специалисту при передаче необходимо вручную сверяет коды образцов в контейнере с ведомостями, что при больших объемах данных становится крайне неэффективным. Процесс перемещения образцов между лабораториями также не фиксируется должным образом, что затрудняет их отслеживание и управление. Отсутствие систематизированного учета времени, проведенного образцами в лабораториях, дополнительно снижает продуктивность и создает сложности в управлении данными.

С учетом стремления лабораторий к автоматизации рутинных задач и повышению эффективности работников, разработка современных решений на основе технологий распознавания изображений и систем учета становится особенно востребованной. Внедрение автоматизированной системы позволит снизить временные затраты, улучшить контроль над перемещением образцов и повысить качество исследований.

Основными **проблемами**, с которыми сталкиваются лаборатории при работе с керном, являются:

* **Затраты времени.** Проверка 20 образцов вручную занимает около 8 минут, что при больших объемах данных приводит к задержкам и неэффективному использованию ресурсов.
* **Отсутствие систематизированного учета.** Невозможность точного отслеживания времени, проведенного образцами в лабораториях, затрудняет анализ и планирование.
* **Сложности отслеживания перемещений образцов.** Часто перемещения происходят хаотично, без четкой фиксации, что осложняет управление процессом и контроль за состоянием образцов.

Эти проблемы вызывают потребность в создании программного обеспечения, способного автоматизировать ключевые процессы обработки, учета и контроля образцов.

**Целью** является создание системы, которая объединит функции распознавания образцов керна по фотографиям с использованием сверточных нейронных сетей и OCR-технологий, а также предоставит веб-интерфейс для управления данными.

Это решение должно обеспечить повышение точности и скорости обработки данных, оптимизацию процессов проверки образцов и улучшение систематизированного учета их состояния.

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

1. Провести анализ требований пользователей (эксперты лаборатории, менеджеры).
2. Изучить существующие решения детекции и сегментации объектов на изображениях.
3. Собрать и подготовить датасет для распознавания керна и текста с использованием YOLOv8.
4. Реализовать модуль распознавания керна и текста на изображениях (YOLOv8).
5. Разработать модуль предобработки изображений.
6. Реализовать модуль распознавания текста (EasyOCR).
7. Разработать веб-приложение для управления данными (React + FastAPI).
8. Объединить все модули в единую систему.

# ГЛАВА 1. ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

## 1.1 ВВЕДЕНИЕ В ПРЕДМЕТНУЮ ОБЛАСТЬ

Предметной областью данной системы является автоматизация процессов отслеживания образцов керна в лабораторных условиях. Образцы керна представляют собой цилиндрические вырезки горной породы, извлеченные из скважин для последующего проведения геологических исследований. Эти образцы используются для анализа структуры и состава породы, что имеет ключевое значение для оценки потенциала месторождений.

На каждом этапе лабораторных исследований специалист осматривает образцы и сверяется с ведомостью, которая прилагается вместе с образцами, или с сообщением электронной почты, которое пришло ему из предыдущей лаборатории. Весь процесс сверки происходит полностью вручную и занимает много времени, особенно в обработке больших партий образцов.

Кроме того, перемещение образцов между лабораториями часто не фиксируется должным образом, что затрудняет их отслеживание и управление. Отсутствие систематизированного учета керна и времени, проведенного образцом в лаборатории, также снижает эффективность работы и приводит к сложности в управлении данными.

Таким образом, автоматизация этих процессов с использованием современных технологий, таких как нейронные сети для распознавания изображений и построения маршрутов перемещения образцов, позволит значительно улучшить точность, скорость и эффективность работы лабораторий, а также повысить качество контроля за состоянием образцов.

## 1.2 АНАЛИЗ СУЩЕСТВУЮЩИХ ПОДХОДОВ

В рамках исследования был проведён анализ существующих подходов к решению схожих задач. Результаты анализа показали, что многие исследователи и компании уже сталкивались с аналогичными проблемами, связанными с автоматизацией обработки образцов керна. Однако каждое из существующих решений имеет свои ограничения, которые влияют на эффективность их применения.

Для структурированного анализа выделено несколько ключевых признаков, по которым сравниваются подходы:

**1. Формат входных данных**

Существенное различие между существующими решениями заключается в типе и качестве обрабатываемых изображений. Входные данные могут различаться следующим образом:

* **Формат съемки:**
  + Камеры, фиксирующие отдельные образцы керна.
  + Камеры, фотографирующие целые коробки с образцами.
* **Качество изображения:**
  + Фотографии, сделанные вручную, что может привести к вариациям в освещении, высоте съемки, наличию посторонних объектов.
  + Изображения, полученные со статичных камер или роботизированных систем, которые обеспечивают стабильные условия съемки. Эти различия влияют на сложность последующей обработки данных, включая этапы предобработки и обучения моделей.

#### **2. Доступность решений**

Инструменты, используемые для автоматизации анализа, также имеют значительные отличия:

* **Стоимость:** Некоторые решения являются коммерческими продуктами с высокой стоимостью, что ограничивает их доступность.
* **Коммерческая тайна:** Многие подходы недоступны для покупки или изучения из-за закрытости технологий и алгоритмов.
* **Открытые решения:** Существующие открытые решения зачастую оказываются недостаточно надежными или функциональными для решения задач обработки керна. Эти факторы ограничивают применение готовых решений и подчеркивают необходимость разработки новых методов.

#### **3. Подходы к маркировке образцов керна**

Еще одним ключевым различием является способ нанесения кодов на образцы, который напрямую влияет на их распознаваемость системами:

* **Печатные шрифты:** Используются печатные роботы и машинки, что обеспечивает стандартизированный вид маркировки.
* **Ручная маркировка:** Коды наносятся вручную, что приводит к значительным вариациям в почерке, размере и читаемости символов. Разнообразие методов маркировки усложняет унификацию алгоритмов распознавания и требует адаптивных подходов в разработке системы.

Анализ показал, что существующие подходы обладают следующими ограничениями:

* Разнообразие форматов входных данных требует гибкости и устойчивости алгоритмов обработки.
* Доступность инструментов ограничена из-за высокой стоимости или закрытости технологий.
* Различия в методах маркировки требуют универсальных подходов к распознаванию текстов и символов.

Одним из примеров успешного применения современных технологий в задачах автоматизации маркировки керна является разработка модуля идентификации для роботизированной системы *[Сухов, Шакин, с. 27–32]*. Исследователи использовали комбинацию методов компьютерного зрения, библиотек OpenCV *[OpenCV Documentation]* и Tesseract-OCR для считывания текста с образцов. Ключевой особенностью подхода стало использование нейронной сети YOLOv5 для предварительного выделения областей с текстом, что значительно повысило точность распознавания. Недостатками являются устаревшие технологии YOLOv5 (разработана в 2020 году) и Tesseract-OCR (разработан в 2005 году, эффективен преимущественно для распознавания печатного текста).

Рассмотрим метод автоматической коррекции угла поворота изображения, описанный в статье Нецепляева Д. *[Нецепляев]*, оказался неподходящим для решения поставленной задачи. В этом подходе изображение поворачивается на малые углы с использованием алгоритма, включающего бинаризацию, уменьшение изображения, определение контуров и аффинное преобразование для коррекции угла. Однако этот метод не был эффективен в нашем случае, поскольку требуемые углы поворота изображения (более 45 градусов) значительно превышают возможности предложенной технологии.

В статье компании "Инфосистемы Джет" *[Камшуков]* рассматривается задача распознавания серийных номеров на металлических брусках с произвольной ориентацией. Среди протестированных OCR-решений, таких как PaddleOCR, Tesseract и EasyOCR, наилучшие результаты продемонстрировала EasyOCR *[EasyOCR Репозиторий проекта]*, особенно после обработки изображений, включая бинаризацию и инверсию. Для улучшения распознавания текста предложен подход с предварительным поворотом изображения. Алгоритм включает сегментацию области текста, построение прямой с помощью линейной регрессии для определения угла наклона и дальнейший поворот текста в горизонтальную ориентацию. Этот подход позволил значительно повысить точность работы моделей, особенно на сложных и специфичных данных. Этот подход стал основой для выбора модели EasyOCR и разработки собственного алгоритма поворота текста на изображениях.

Таким образом, на основании проведённого анализа можно сделать вывод, что существующие решения не в полной мере удовлетворяют требованиям автоматизации процесса обработки образцов керна для предприятия Новатэк НТЦ. Это подчеркивает актуальность разработки новой системы, способной преодолеть указанные ограничения.

## 1.3 ИДЕЯ РЕШЕНИЯ

Основная идея заключается в создании системы, которая отслеживает перемещение образцов керна в лабораториях. Решение предполагает использование технологий глубокого обучения для обработки изображений и веб-интерфейса для управления и анализа данных.

Ключевые задача, которую будет решать наша система:

Распознавание кодов образцов керна. Это позволит автоматизировать процесс составления списков образцов для ведомостей и сверки данных.

Для реализации системы будут использованы три нейронные сети:

* Первая сеть будет определять местоположение образцов на изображении и вырезать их из общего фона.
* Вторая сеть будет ориентированно распознавать местоположение текста на образце для дальнейшего поворота изображения.
* Третья сеть будет распознавать коды образцов, написанные на торце керна.

Результатом станет веб-приложение, которое позволит загружать фотографии образцов,ведомости, автоматически обрабатывать их с помощью обученных моделей.

## 1.4. ТРЕБОВАНИЯ И ОГРАНИЧЕНИЯ СИСТЕМЫ

Разработка системы распознавания и учета образцов керна основывается на ряде функциональных и нефункциональных требований, а также на технических и эксплуатационных ограничениях. Эти требования были определены на основе анализа текущих процессов лабораторий и задач, которые необходимо автоматизировать. Дополнительно в проекте учитывались технические ограничения, связанные с доступностью оборудования и программного обеспечения.

Один из важных факторов, влияющих на выбор технологий, связан с ограничениями региона: актуальная версия CUDA заблокирована, что вынуждает использовать не последнюю версию. Это накладывает ограничения на использование современных моделей глубокого обучения. Наилучший компромисс между производительностью и доступными ресурсами был достигнут при выборе YOLOv8, поскольку она одна из самых эффективных моделей, поддерживаемых используемой версией CUDA.

**Функциональные требования**

1. **Проверка данных**: Возможность сравнения автоматически распознанных кодов образцов с загруженными списками из документов (например, Excel) для проверки корректности данных.
2. **Интерфейс пользователя**: Веб-интерфейс, предоставляющий возможность:
   * загрузки изображений образцов керна;
   * загрузки ведомостей с кодами образцов;
   * выбора лаборатории, откуда поступили образцы;
   * последующего анализа данных.

**Нефункциональные требования**

**Производительность**: Программная обработка изображения для 20 образцов должна занимать не более 1 минуты.

**Ограничения**

1. **Качество изображений**: Все фотографии образцов керна должны быть сделаны с использованием камеры с разрешением не ниже 4K.
2. **Освещение**: Фотографии должны быть выполнены при хорошем освещении, чтобы минимизировать ошибки при распознавании кодов.
3. **Фон**: Задний фон изображения должен быть однотонным, без посторонних предметов, и большая часть кадра должна занимать коробка с образцами.
4. **Аппаратное и программное обеспечение**: Использование современных моделей ограничено из-за недоступности актуальной версии CUDA в регионе. Выбор модели YOLOv8 обусловлен её эффективностью при использовании поддерживаемой версии CUDA.

Этот набор требований и ограничений обеспечивает основу для проектирования системы, которая соответствует потребностям лабораторий и учитывает существующие технические ограничения.

## 1.5 ТЕОРЕТИЧЕСКОЕ ОБОСНОВАНИЕ ТЕХНОЛОГИЙ

Проект основан на использовании современных методов и алгоритмов обработки изображений и анализа данных.

**Основные теоретические положения:**

1. **Компьютерное зрение:**

Для автоматической идентификации образцов керна и текста используются нейронные сети, решающие задачи детекции объектов и распознавания текста. Это позволяет выделять изображения образцов на фоне коробки и извлекать коды с торцов керна.

1. **Распознавание текста (OCR):**

Используется алгоритм Optical Character Recognition (OCR), основанный на сверточных и рекуррентных нейронных сетях, для преобразования текстовой информации с изображений в машиночитаемый формат.

1. **Предварительная обработка изображений:**

Для повышения точности работы нейронных сетей используются методы аффинных преобразований, нормализации и контрастирования (CLAHE). CLAHE (Contrast Limited Adaptive Histogram Equalization) — это метод улучшения контраста, который работает путем локального выравнивания гистограммы на небольших участках изображения. Такой подход позволяет избежать усиления шума в однородных областях и улучшает качество деталей, что делает изображения более подходящими для обработки нейронными сетями.

**Теоретическая база для обучения нейронных сетей:**

* **Функции потерь:** Для обучения сетей используются специализированные функции потерь, такие как Cross-Entropy Loss для классификации и CTC Loss для распознавания текста.
* **Архитектуры нейронных сетей:  
   YOLO (You Only Look Once)** — это семейство архитектур свёрточных нейронных сетей для детекции объектов в реальном времени. YOLOv8 является одной из последних версий и включает улучшенные механизмы детекции объектов с использованием адаптивных анкоров, внимания (attention) и более глубоких слоев сети.  
  Модель обучается на больших наборах данных, содержащих разметку с координатами объектов (bounding boxes). Типичные датасеты включают COCO (Common Objects in Context) и специализированные наборы данных с аннотациями, созданными для решения задач конкретной предметной области.  
  В контексте проекта YOLOv8 была дообучена на выборке изображений керна и коробок, где вручную размечены координаты кернов и их текстовых меток.

**EasyOCR** *[EasyOCR Репозиторий проекта]* — это библиотека для распознавания текста, построенная на основе сверточных (CNN) и рекуррентных нейронных сетей (LSTM) с использованием механизма внимания (Attention Mechanism). Она обучена на большом количестве текстовых данных различных языков и шрифтов. Для повышения точности в проекте EasyOCR может быть дообучена на изображениях, содержащих специфические форматы и шрифты, используемые для маркировки кернов.

### 

# ГЛАВА 2. РЕАЛИЗАЦИЯ СИСТЕМЫ

## 2.1 ПОСТАНОВКА ЗАДАЧИ

Для реализации системы автоматизации процессов работы с образцами керна необходимо разработать решение, включающее функциональные модули для распознавания, учета и управления данными. Система должна быть ориентирована на работу в лабораторной среде и решать следующие задачи:

1. **Распознавание образцов керна**
   * Реализовать модуль на основе модели YOLOv8, адаптированной для детекции образцов керна на изображениях.
   * Обеспечить возможность детекции объектов при различных углах съемки и условиях освещения.
   * Использовать алгоритмы предобработки изображений (например, CLAHE) для улучшения качества входных данных.
2. **Распознавание текстовой информации**

Внедрить OCR-модуль с использованием EasyOCR для распознавания кодов образцов.

1. **Веб-интерфейс для управления системой**

Создать веб-приложение на основе FastAPI и React, включающее:

* + - Страницу загрузки изображений и ведомостей с кодами образцов.
    - Возможность проверки результатов распознавания.

1. **Производительность и масштабируемость**
   * Оптимизировать производительность системы для обработки 20 образцов за 1 минуту.
   * Разработать архитектуру системы, обеспечивающую её масштабируемость и возможность работы с большими объемами данных.
2. **Технические ограничения**
   * Учесть, что проект разрабатывается с использованием CUDA 11.2, что ограничивает выбор моделей глубокого обучения.
   * Обеспечить совместимость с камерами, снимающими изображения с разрешением не ниже 4K, при соблюдении условий однотонного фона и хорошего освещения.
3. **Тестирование и верификация**
   * Организовать процесс тестирования для оценки точности работы системы.
   * Провести оптимизацию и исправление выявленных недостатков на этапе тестирования.

Данная постановка задачи служит основой для дальнейшей реализации системы, обеспечивающей автоматизацию работы с образцами керна в лабораторной среде.

## 2.2 ОПИСАНИЕ ИСПОЛЬЗУЕМЫХ ТЕХНОЛОГИЙ

Проект реализуется с использованием современных технологий и инструментов, обеспечивающих надежность и масштабируемость системы. Ниже приведены используемые технологии и инструменты.

**Библиотеки и фреймворки:**

* **Torch**: Библиотека для работы с нейронными сетями, обеспечивающая выполнение вычислений на GPU и поддержку глубокого обучения.
* **Ultralytics** []: Инструмент, позволяющий интегрировать модель YOLOv8 для задач детекции объектов, таких как распознавание образцов керна и текста на изображениях.
* **Cv2 (OpenCV)**: Мощный инструмент для обработки изображений, включающий нормализацию, улучшение контраста, удаление шумов и геометрические преобразования.
* **Matplotlib**: Используется для визуализации результатов обработки изображений и детекции объектов.
* **Numpy**: Библиотека для высокопроизводительных вычислений и работы с многомерными массивами данных.
* **Pillow**: Простая и удобная библиотека для загрузки, обработки и сохранения изображений.
* **sklearn**: Набор инструментов для машинного обучения, используемый, например, для кластеризации данных методом KMeans.

**Дополнительные инструменты:**

* **logging**: Библиотека для регистрации и анализа событий в системе, что облегчает диагностику и исправление ошибок.

**Средства веб-разработки:**

* **FastAPI**: Быстрый и удобный фреймворк для создания REST API, обеспечивающий взаимодействие между клиентской и серверной частями системы.
* **React**: Популярная библиотека для создания динамичных и интуитивно понятных пользовательских интерфейсов.

#### **Аннотирование данных:**

Для создания и разметки датасета использовался инструмент CVAT (Computer Vision Annotation Tool), который был развернут локально. CVAT позволяет удобно аннотировать изображения, создавая рамки, полигональные области или маски для обучения моделей глубокого обучения. Его локальное развертывание обеспечивает высокую производительность и безопасность данных, позволяя работать без необходимости подключения к внешним серверам.

**Подходы к разработке и организации кода:**

* **Модульность:** Код разделен на независимые компоненты, что упрощает тестирование, обновление и добавление нового функционала.
* **Повторное использование кода:** Использование функций и классов, которые можно многократно применять в различных модулях проекта, минимизируя дублирование логики.
* **Документированность:** Каждый модуль содержит комментарии и документацию, упрощающие понимание структуры и логики работы системы.

**Практическое применение технологий:**

* **Предобработка изображений:**

Удаление шума, улучшение контрастности и выделение ключевых объектов для повышения качества детекции. А также использование алгоритма CLAHE из библиотеки OpenCV для улучшения контраста.

* **Детекция объектов:**

Автоматическое выделение образцов керна и текстовых меток с помощью модели **YOLOv8**, адаптированной для работы с ориентированными рамками (Oriented Bounding Box).

* **Кластеризация:**

Применение алгоритма **KMeans** из библиотеки **sklearn** для анализа распределения объектов и разделения данных на группы.

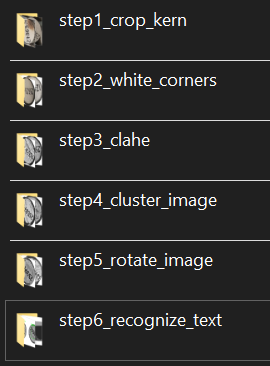
## 2.3 ОПИСАНИЕ ФАЙЛОВОГО ХРАНИЛИЩА

Файловое хранилище представляет собой систему папок, предназначенную для хранения промежуточных и итоговых результатов работы системы. Оно необходимо для диагностики и анализа работы каждого блока Pipeline в отдельности. Результаты обработки Pipeline передаются на Frontend для визуализации работы системы. Кроме того, в файловом хранилище сохраняются все изображения, отправленные на обработку через веб-интерфейс.

**Структура и назначение папок:**

* **step\_1**: Содержит изображения образцов керна, обнаруженные моделью.
* **step\_2**: Хранит изображения с удалённым шумом, очищенные от лишних элементов на углах.
* **step\_3**: Содержит изображения с улучшенной контрастностью для повышения качества обработки.
* **step\_4**: Хранит кластеризованные изображения, сгруппированные по определённым признакам.
* **step\_5**: Содержит изображения с текстом, ориентированным горизонтально для дальнейшего распознавания.
* **step\_6**: В этой папке находятся изображения с визуализированными результатами распознавания текста.

Такой подход к организации файлового хранилища обеспечивает прозрачность работы системы и удобство анализа её эффективности.



*Рис. 1. Структура файлового хранилища*

## 2.4 РАЗРАБОТКА НЕЙРОННЫХ СЕТЕЙ

### 2.4.1 МОДЕЛЬ YOLO. ДЕТЕКЦИЯ ОБРАЗЦОВ КЕРНА

Для обучения модели YOLO был создан собственный датасет. Образцы керна фотографировались вручную, а разметка данных выполнялась с использованием сервиса **CVAT (Computer Vision Annotation Tool)**. Итоговый датасет включает 58 изображений с 1900 аннотациями. На изображениях отображены коробки с керном. Пример данных (Рис. 2)

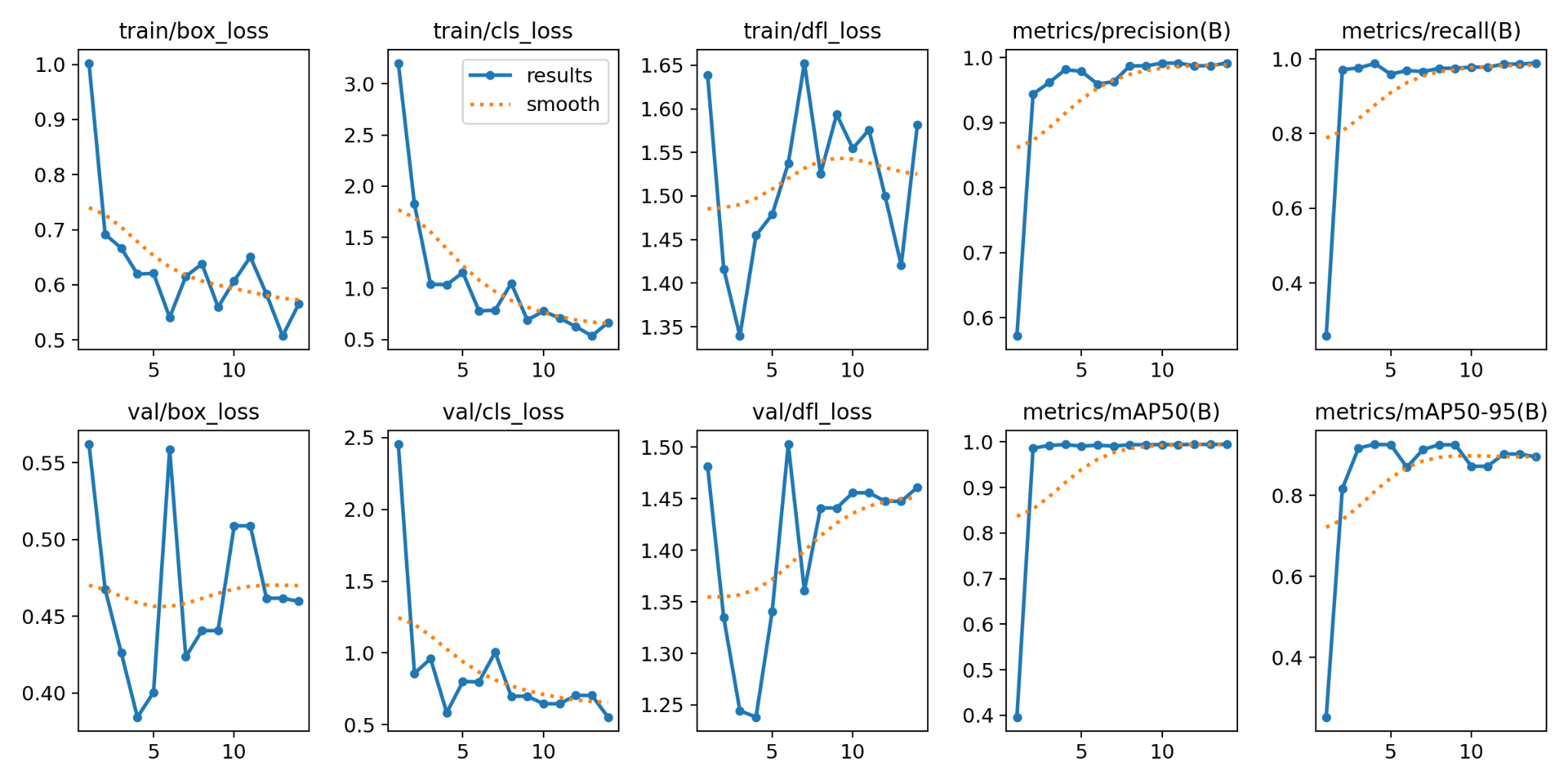
**

*Рис. 2. Фотография коробки с керном.*

Изначально модель была предобучена на 20 изображениях. После этого модель использовалась для автоматической разметки оставшихся изображений, и полученные аннотации вручную корректировались. Обучение проводилось на серверной видеокарте **Tesla P40**. Процесс завершился на 15-й эпохе благодаря функции автоматической остановки библиотеки **Ultralytics**, которая предотвращает переобучение. Общее время обучения составило 7 минут.В тестовой выборке из 280 аннотаций модель корректно определила 244 объекта. Результаты модели:

* Precision и Recall достигают 100% с точностью eps = 10⁴.
* mAP 50–95 ≈ 87%.

С подробными результатами обучения можно ознакомиться на графиках (Рис. 3).



*Рис. 3. Графики метрик обучения модели детекции керна*

### 2.4.2 МОДЕЛЬ YOLO. ДЕТЕКЦИЯ ТЕКСТА

Датасет для детекции текста был сформирован с использованием результатов предыдущей модели, обученной на детекцию керна, в рамках 4-го шага Pipeline. Разметка данных также проводилась с использованием **CVAT**. Датасет включает 190 изображений с 190 аннотациями. Пример изображения датасета (Рис. 4):

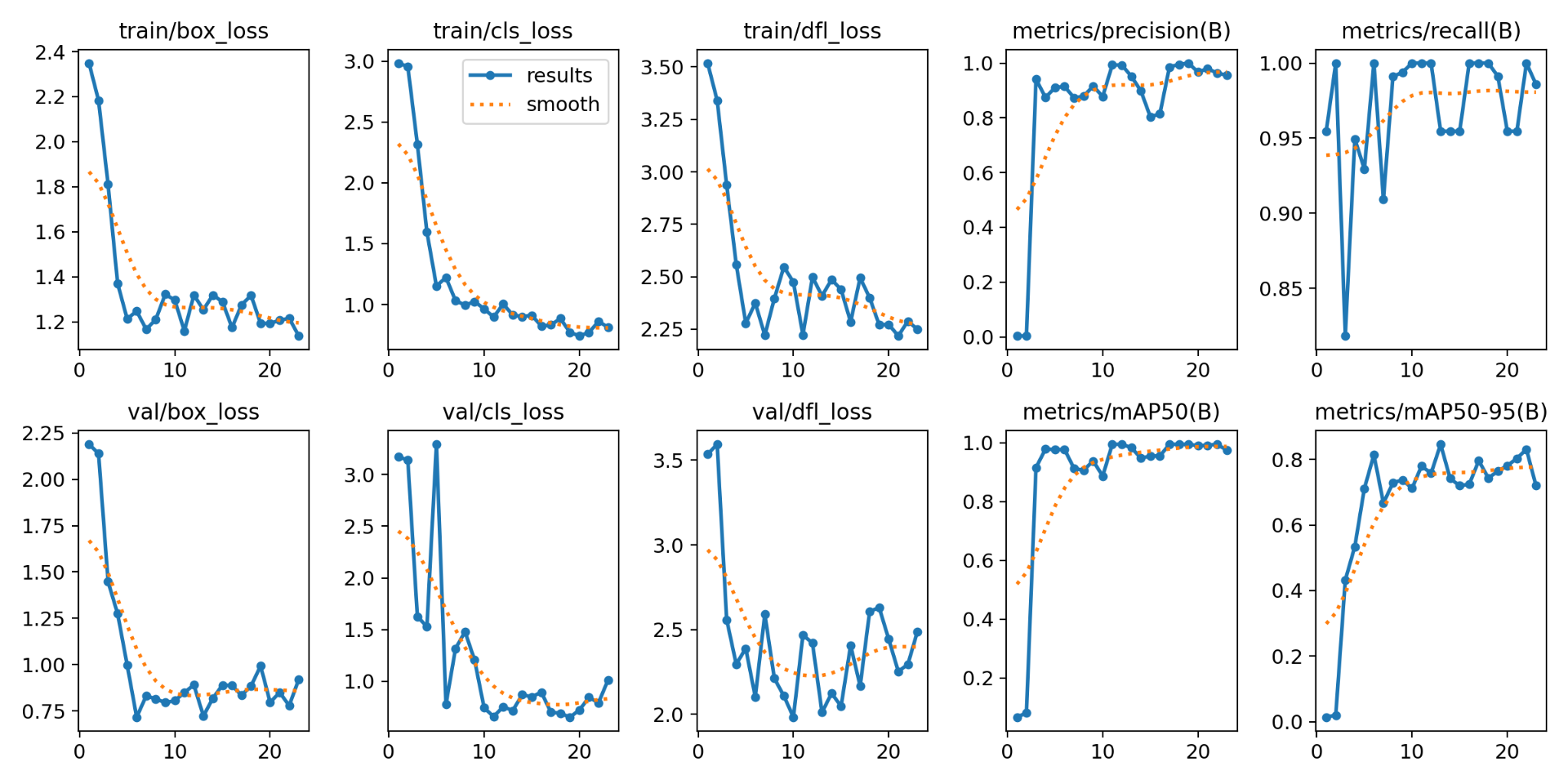


*Рис. 4. Изображения керна после кластеризации K-means*

Обучение проводилось на той же серверной видеокарте **Tesla P40 (24GB)**. Процесс завершился на 15-й эпохе благодаря автоматической остановке библиотеки **Ultralytics**. Время обучения составило 4 минуты.В тестовой выборке из 29 аннотаций модель корректно определила 22 объекта. Получились такие результаты:

* Precision ≈ 90%
* Recall ≈ 97%.
* mAP 50–95 ≈ 70%.

С подробными результатами обучения можно ознакомиться на графиках (Рис. 5).С результатами модели можно ознакомиться в *[Таблица 1]*.



*Рис. 5. Графики метрик обучения модели детекции керна*

### 2.4.3 МОДЕЛЬ EASYOCR. РАСПОЗНАВАНИЕ ТЕКСТА

Для задачи распознавания текста использовалась предобученная модель **EasyOCR**, обученная на широком наборе данных, включающем изображения текста на различных языках, шрифтах и форматах. Источники данных включают открытые датасеты, такие как **ICDAR** (International Conference on Document Analysis and Recognition) и **SynthText** (синтетически сгенерированные текстовые изображения). С результатами модели можно ознакомиться в *[Таблица 1]*.

## 

## 2.5 РАЗРАБОТКА ПРИЛОЖЕНИЯ

### 2.5.1 ХОД РАБОТЫ

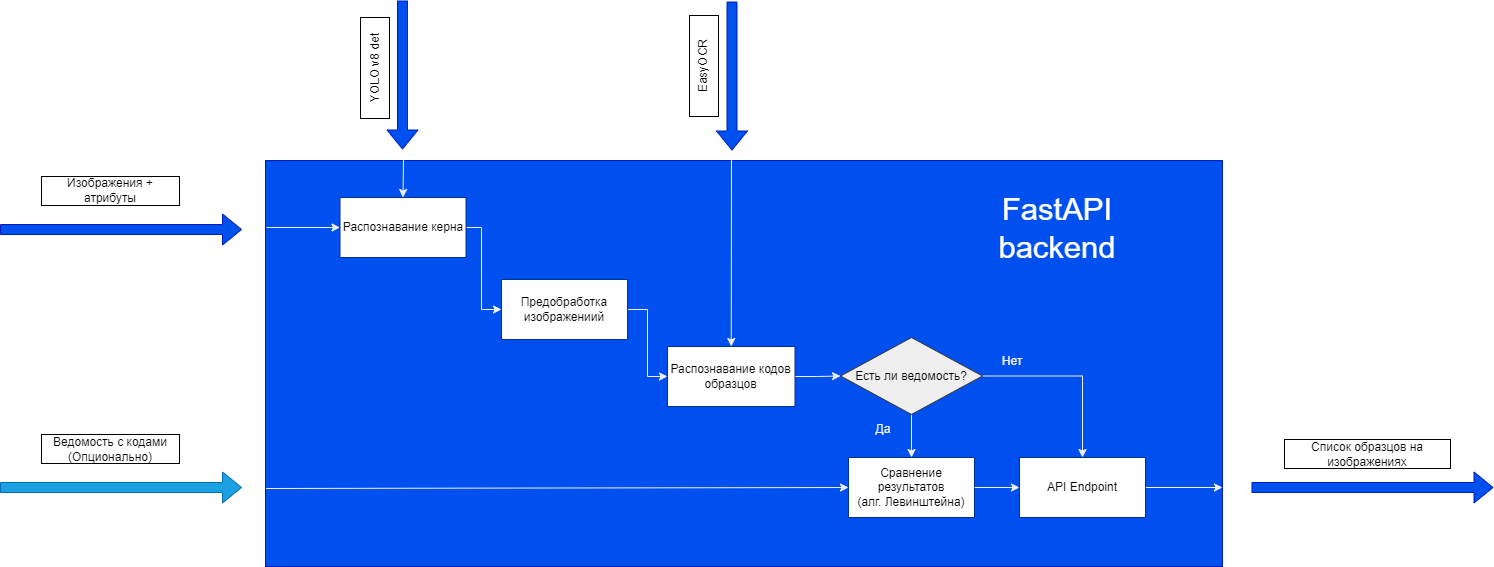
В рамках работы была реализована автоматизированная система для распознавания и учета образцов керна, объединяющая нейронные сети, модули предобработки и веб-приложение. Основные этапы разработки включают следующие шаги:

1. Сбор изображений керна и текста.
2. Разметка данных с использованием CVAT для модели детекции керна.
3. Обучение YOLOv8 для детекции керна.
4. Исследование видов предобработки изображений
5. Определение оптимального порядка операций.
6. Обработка результатов модели детекции керна.
7. Разработка алгоритма окрашивания углов в белые пиксели.
8. Интеграция алгоритмов K-means и CLAHE.
9. Разметка данных с использованием CVAT для модели детекции текста.
10. Обучение YOLOv8 для детекции текста.
11. Разработка алгоритма поворота текста.
12. Интеграция распознавания текста EasyOCR.
13. Разработка интерфейса на React.
14. Реализация API на FastAPI.
15. Объединение модулей в единую систему

Наглядное отображение этих шагов можно увидеть в нашей архитектуре приложения

### 2.5.2 ФУНКЦИОНАЛЬНАЯ СХЕМА

Функциональная схема (Рис. 6): проекта представляет собой визуализацию ключевых этапов работы системы и их взаимодействия. Она отображает основные модули, последовательность их выполнения и данные, которые передаются между ними.



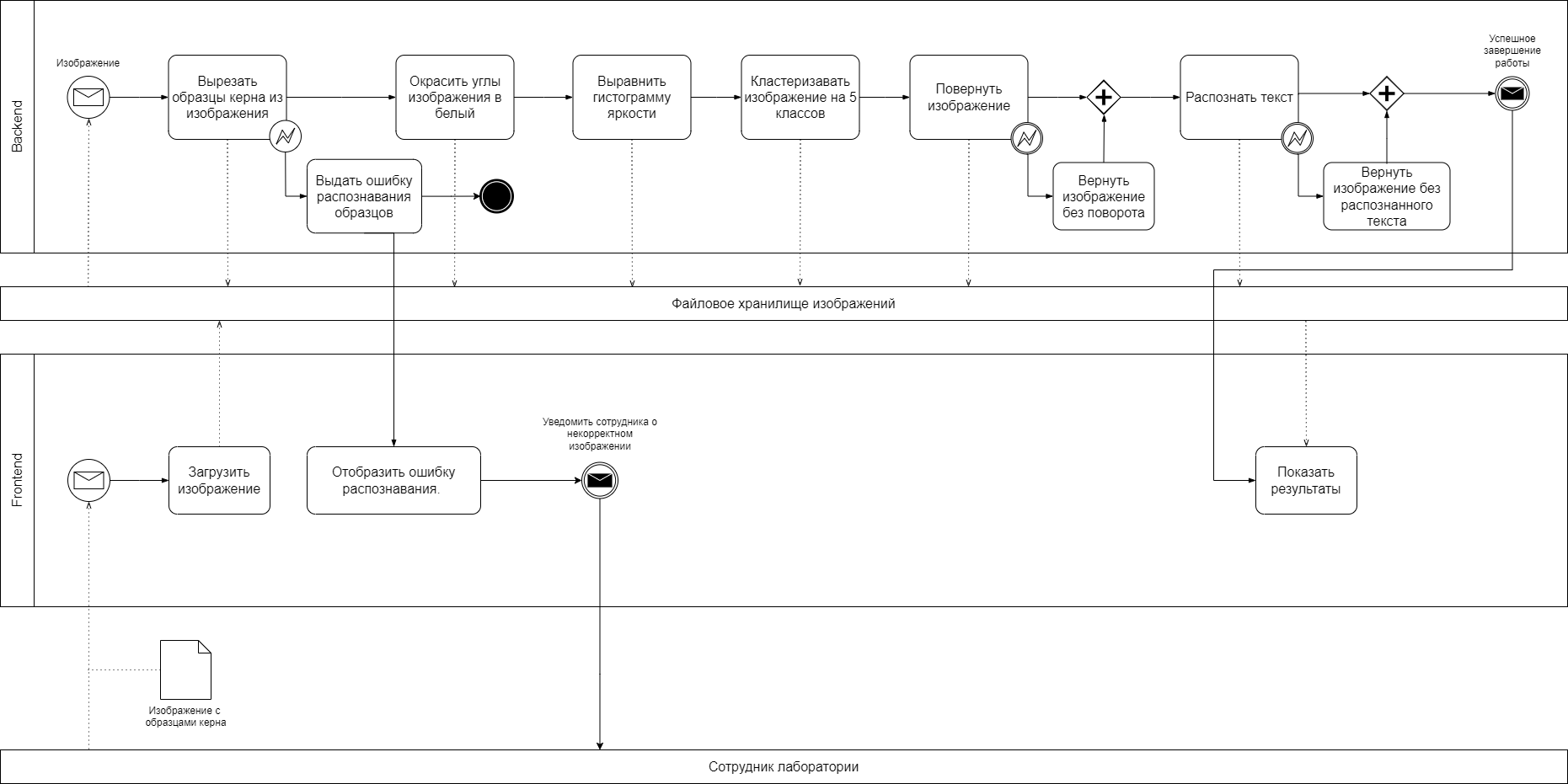
*Рис. 6. Функциональная схема серверной части*

**Поток выполнения**

1. **Входные данные:**
   * Изображения образцов с атрибутами (передаются в систему для обработки).
   * Ведомость с кодами образцов (необязательный элемент, используется для проверки распознанных данных).
2. **Обработка изображений:**
   * **Распознавание керна:** С помощью модели YOLOv8 выполняется определение областей на изображении, где находятся керны.
   * **Предобработка изображений:** Найденные области передаются для подготовки данных к дальнейшему распознаванию (например, улучшение контраста или поворот).
3. **Распознавание кодов образцов:**
   * Используется библиотека EasyOCR для извлечения текстовой информации (кодов) из подготовленных изображений.
4. **Проверка наличия ведомости:**
   * Если ведомость предоставлена, происходит сравнение результатов распознавания с её данными. Для этого используется алгоритм Левенштейна, позволяющий оценить совпадение текстов.
5. **Выходные данные:**
   * В случае отсутствия ведомости формируется список образцов и их кодов на основе анализа изображений.
   * Если ведомость предоставлена, выдаётся результат сверки: корректные и некорректные распознанные данные.
6. **Интеграция через API Endpoint:**
   * Все обработанные данные доступны через API, обеспечивая взаимодействие системы с веб интерфейсом.

### 2.5.3 АРХИТЕКТУРА СИСТЕМЫ

Система направлена на автоматизацию обработки изображений образцов керна с использованием нейронных сетей. Архитектура включает несколько взаимосвязанных компонентов, каждый из которых выполняет свою роль. С представлением архитектуры в нотации BPMN (Business Process Model and Notation) можно ознакомиться ниже (Рис. 7):



*Рис. 7. Архитектура в нотации BPMN*

Система состоит из основных компонентов:

* **Frontend (веб-интерфейс)** – клиентская часть на React, позволяющая пользователю загружать изображения, отслеживать прогресс обработки и получать результаты в виде текстовых данных.
* **Backend (серверная логика)** – сервер на FastAPI, который обрабатывает изображения. Он выполняет задачи, такие детекция керна, улучшение качества, кластеризация и распознавание текста.
* **Pipeline (конвейер обработки изображений)** – связывает все модули системы, организуя последовательную обработку изображений, включая:
  + Детекция образцов керна с последующим вырезанием.
  + Удаления информации о углах изображения (перекраска пикселей в белый цвет)
  + Улучшение контраста с помощью алгоритма CLAHE.
  + Кластеризация для определения границ.
  + Поворот изображения осуществляется с использованием детекции текста и аффинного преобразования.
  + Распознавание текста с помощью нейронных сетей.
* **Пользователь** – взаимодействует с веб-интерфейсом, загружая изображения, отслеживая их обработку и получая результаты распознавания текста и другие данные.

**Взаимосвязь компонентов**

Процесс начинается с того, что **пользователь** загружает изображение через frontend. Оно отправляется на сервер, где backend обрабатывает изображение через Pipeline. Результаты промежуточной обработки сохраняются в файловом хранилище, что позволяет отслеживать прогресс и повторно использовать данные на любом этапе.

Каждый шаг (например, улучшение качества через CLAHE или детекция зерен) выполняется отдельным модулем. После завершения обработки результаты отправляются в frontend для отображения финальных изображений и распознанного текста, которые становятся доступными пользователю.

**Организация работы**

Процесс обработки состоит из последовательных шагов с независимыми модулями, что обеспечивает гибкость системы. Модульная структура позволяет легко добавлять новые этапы или улучшать существующие. Взаимодействие между frontend и backend упрощает пользовательский опыт, а Pipeline гарантирует корректность обработки.

### 2.5.4 ОПИСАНИЕ МОДУЛЕЙ

Для реализации функциональности системы разработаны отдельные модули, каждый из которых выполняет свою задачу. Такая модульная структура позволяет повысить читаемость, масштабируемость и удобство тестирования кода. Ниже приводится описание ключевых модулей, используемых в проекте:

* **FolderOperation**Модуль, обеспечивающий управление папками. Его основная задача — очистка выходных директорий на начальном этапе работы PipeLine, что предотвращает накопление ненужных данных. Кроме того, модуль используется для создания датасета с помощью автоаннотирования, основанного на результатах предобученной модели.
* **ImageOperation**Модуль для обработки изображений. Включает функции удаления шума (например, окрашивание углов в белый цвет), повышения качества изображения с использованием алгоритма CLAHE, а также методы кластеризации для группировки данных.
* **KernDetection**Модуль для работы с моделью детекции образцов керна и текстовых областей на изображениях. Содержит промежуточные методы извлечения данных из модели YOLOv8, включая функции для получения необходимой информации и визуализации предсказаний.
* **TextRecognition**Модуль для работы с системой EasyOCR, предназначенной для распознавания текста на изображениях. Дополнительно включает методы для обработки результатов распознавания (например метод сравнения предсказаний с ведомостью) и их визуального представления.

### 2.5.5 ОПИСАНИЕ ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ

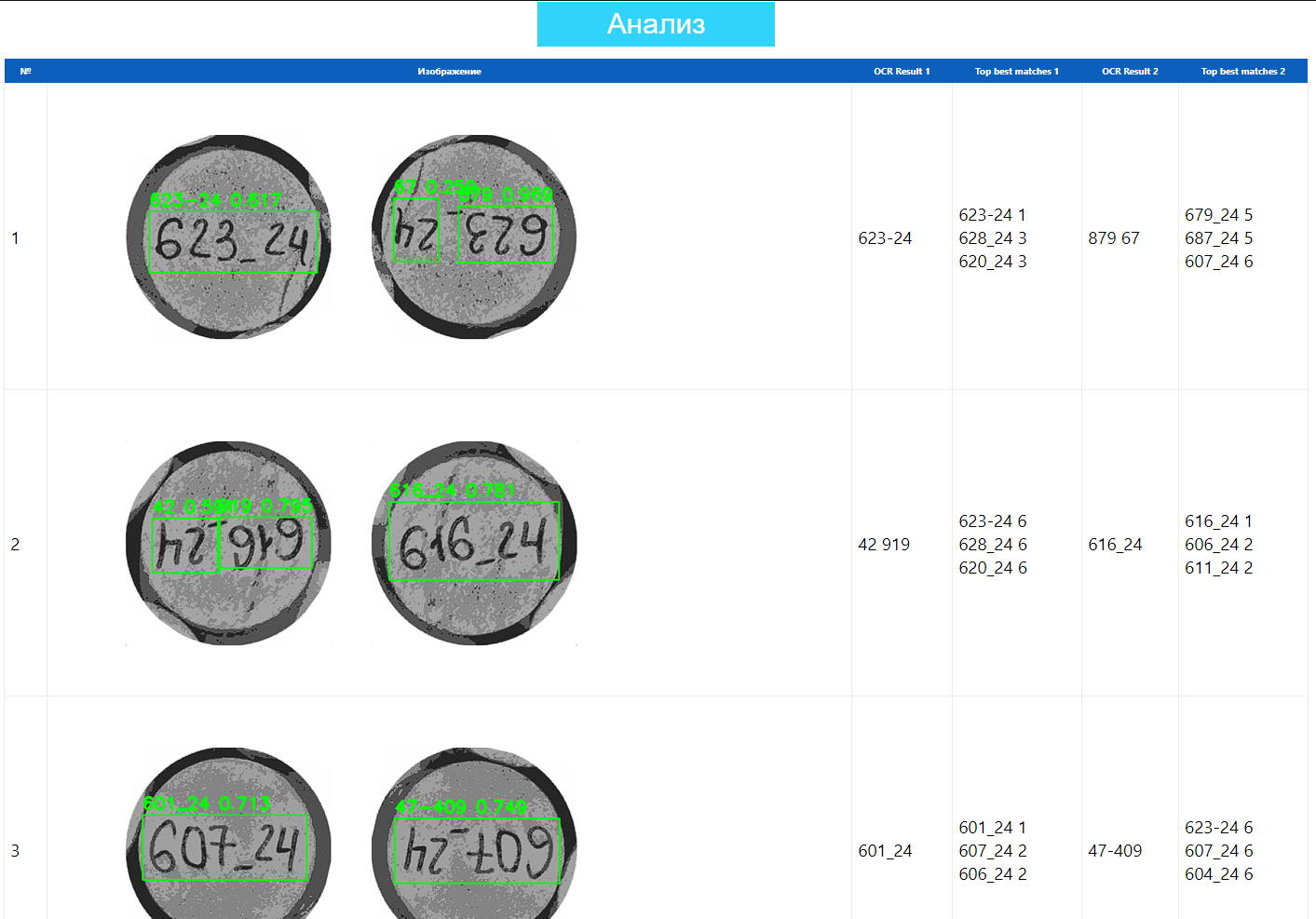
Интерфейс пользователя разработан с учетом требований лаборатории и минимизации усилий, необходимых для взаимодействия с системой. Он обеспечивает доступ ко всем основным функциям проекта.

Главная страница интерфейса предоставляет возможность загружать изображения образцов керна и сопутствующие ведомости, а также выбирать лабораторию, откуда поступили данные (Рис. 8). Это позволяет системе учитывать контекст работы и корректно обрабатывать информацию. После загрузки данных пользователю становится доступна кнопка «Анализ», с помощью которой отправляются запросы на сервер для обработки.

Результаты анализа предоставляются в табличной форме в течение тридцати секунд для 20 образцов. Пользователь может просматривать, анализировать и работать с предоставленными данными (Рис. 9). В дальнейшем интерфейс может быть расширен дополнительными функциями для удобства использования.



*Рис. 8. Главная страница.*



*Рис. 9. Таблица с результатами работы*

## 2.6 ОРГАНИЗАЦИЯ ТЕСТИРОВАНИЯ

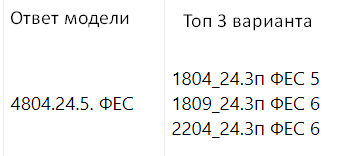
Тестирование включает в себя включает анализ и подсчет метрик каждой модели. Проект включает в себя три модели:

1. Детекция керна (YOLOv8-detection-medium)
2. Детекция текста (YOLOv8-detection-medium)
3. Распознавание текста (Easy-OCR)

Модуль ultralytics (YOLO) включает в себя автоматический сбор метрик модели при обучении. Тестирование заключается в анализе полученных графиков.

Для модели Easy-OCR используется собственная метрика оценки результатов. Тестовые изображения (свыше 100 образцов керна) анализируются с помощью алгоритма Левенштейна. Распознавание считается успешным, если расстояние Левенштейна минимально и результат входит в топ-3 наиболее близких вариантов.

Для образца с кодом 1804\_24.3п ФЕС (Рис. 10) модель выдала ответ 4804.24.5. ФЕС. Среди вариантов из актов найден верный код с расстоянием Левенштейна равным 5. Такой случай считается успешным.



*Рис. 10. Пример работы распознавания текста*

Даже если расстояние Левенштейна одинаково (например, равно 3 для нескольких вариантов), результат 1816\_24.3п ФЕС считается верным (Рис. 11), если он попадает в топ-3.



*Рис. 11. Пример работы 2 распознавания текста с одинаковым расстоянием Левенштейна*

**Расчет метрики EasyOCR**

Подсчитывается количество успешных и неуспешных распознаваний (1), на основе чего определяется точность этой модели.



На момент тестирования (15.12.2024) зафиксированы результаты *[Таблица 1]*:

1. Достигнуто распознавания керна на изображении с точностью 98,4%
2. Достигнуто распознавания местоположения текста изображения с точностью 93%
3. Достигнуто распознавание текста с точностью 66,7%
4. Разработан веб-интерфейс с загрузкой изображений и сравнением предсказанных результатов.

Тестирование системы показало, что текущая версия продукта не решает проблему учета и отслеживания образцов в лабораториях, так как данный функционал еще не реализован. Он планируется к разработке в следующих версиях системы.

Также, точность распознавания текста, составляющая 66,7%, недостаточна для обеспечения значительного ускорения процесса формирования списков в работе лабораторий. Для достижения требуемой производительности и точности в будущем потребуется улучшение алгоритмов распознавания текста и дополнительных доработок системы.

Таким образом, несмотря на успешную реализацию некоторых ключевых функций, проект еще не полностью решает поставленные задачи и требует доработок для обеспечения полного выполнения функционала.

# ЗАКЛЮЧЕНИЕ

В рамках работы была разработана и реализована система до стадии MVP (минимально жизнеспособного продукта). Основные поставленные задачи выполнены: созданы модули для распознавания образцов керна и текстовой информации на изображениях с использованием нейронных сетей, а также разработан веб-интерфейс, обеспечивающий удобное взаимодействие с системой.

Однако текущая версия системы не решает проблему систематизированного учета образцов, что является одной из ключевых потребностей лабораторий. Этот функционал планируется реализовать в последующих этапах разработки.

Проведенное тестирование показало, что система верно распознает 66,7% образцов. Хотя этот показатель приемлем для этапа MVP, он пока недостаточен для обеспечения значительного ускорения процессов проверки образцов. Это подчеркивает необходимость доработки моделей и алгоритмов для достижения более высокой точности.

Система имеет высокий потенциал для масштабирования и адаптации под изменяющиеся задачи и требования. В будущем планируется:

1. Оставить только один вариант повернутого изображения для улучшения качества сервиса.
2. Дообучить модели детекции керна и текста для улучшения их работы.
3. Развить систему OCR, дообучив EasyOCR для специфических задач проекта.
4. Добавить возможность ввода маски для кода образца для более точного распознавания.
5. Доработать алгоритм последовательного распознавания текста для улучшения качества результатов.
6. Создать базу данных для хранения информации о перемещении образцов, что улучшит их отслеживание и управление.
7. Добавить страницу для верификации результатов предсказания и обзора учета образцов.
8. Работать над ускорением вычислений для повышения эффективности системы.
9. Переписать алгоритм поиска схожих кодов образцов для улучшения точности и скорости поиска.

Таким образом, несмотря на успешную реализацию основной функциональности, проект продолжит развиваться, стремясь к еще более высокому уровню точности.

# 

# СПИСОК ЛИТЕРАТУРЫ

1. Сухов Ф. Я., Шакин В. Ю. Разработка модуля идентификации для системы роботизированной маркировки образцов керна // Лучшая студенческая статья 2023. – С. 27–32. [Электронный ресурс]. – Режим доступа:<https://naukaip.ru/wp-content/uploads/2023/01/K-453.pdf> (дата обращения: 15.12.2024).
2. Нецепляев Д. Автоматическая коррекция угла поворота изображения в задаче распознавания текста // URL:<https://newtechaudit.ru/avtomaticheskaya-korrekcziya-ugla-povorota-izobrazheniya-v-zadache-raspoznavaniya-teksta/> (дата обращения: 15.12.2024).
3. Камшуков И. OCR за час? // Habr. URL:<https://habr.com/ru/articles/656489/> (дата обращения: 15.12.2024).
4. EasyOCR Репозиторий проекта // URL:<https://github.com/JaidedAI/EasyOCR> (дата обращения: 15.12.2024).
5. Python Documentation // URL:<https://docs.python.org/3/> (дата обращения: 15.12.2024).
6. JavaScript Documentation // URL:<https://developer.mozilla.org/en-US/docs/Web/JavaScript> (дата обращения: 15.12.2024).
7. NumPy Documentation // URL:<https://numpy.org/doc/> (дата обращения: 15.12.2024).
8. Pillow Documentation // URL:<https://pillow.readthedocs.io/en/stable/> (дата обращения: 15.12.2024).
9. OpenCV Documentation // URL:<https://docs.opencv.org/> (дата обращения: 15.12.2024).
10. scikit-learn Documentation. KMeans Clustering // URL:<https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html> (дата обращения: 15.12.2024).
11. PyTorch Documentation // URL:<https://pytorch.org/docs/stable/> (дата обращения: 15.12.2024).
12. Ultralytics Documentation // URL:<https://docs.ultralytics.com/> (дата обращения: 15.12.2024).
13. FastAPI Documentation // URL:<https://fastapi.tiangolo.com/> (дата обращения: 15.12.2024).
14. React Documentation // URL:<https://reactjs.org/docs/getting-started.html> (дата обращения: 15.12.2024).
15. Документация CVAT // URL:<https://docs.cvat.ai/docs/> (дата обращения: 15.12.2024).
16. Алгоритм CLAHE // URL: [https://docs.opencv.org/4.x/d5/daf/tutorial\_py\_histogram\_equalization.html](https://habr.com/ru/articles/656489/) (дата обращения: 15.12.2024).

# ПРИЛОЖЕНИЕ

*Таблица. 1.*

*Метрики нейронных сетей на валидационном наборе изображений*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Изображения** | **0001** | **0007** | **0008** |  | **ИТОГО кол-во** | **Точность** |
| Количество образцов | 39 | 30 | 60 |  | 129 |  |
| **YOLO (kern): обнаруженные образцы** | 40 | 31 | 60 |  |  |  |
| **YOLO (kern): ошибки** | 1 | 1 | 0 |  | 2 | **0,984** |
| **YOLO (text): ошибки поворотов** | 3 | 1 | 5 |  | 9 | **0,930** |
| **EasyOCR: ошибки** |  |  |  |  |  |  |
| - неправильная последовательность | 0 | 5 | 15 |  | 20 | **0,845** |
| - распознавание символов | 4 | 9 | 30 |  | 43 | **0,667** |