

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ
КАЗАХСТАН

Некоммерческое акционерное общество «Казахский национальный
исследовательский технический университет имени К.И.Сатпаева»



Институт Автоматики и информационных технологий

Кафедра Робототехники и технических средств автоматки

Юсков Даниил Владимирович

«Исследование методов визуализации и анализа для обнаружения трещин в
трубах»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
к дипломному проекту

6B07111 – Робототехника и мехатроника

Алматы 2025

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ
КАЗАХСТАН

Некоммерческое акционерное общество «Казахский национальный
исследовательский технический университет имени К.И.Сатпаева»



SATBAYEV
UNIVERSITY

Институт Автоматики и информационных технологий

Кафедра Робототехники и технических средств автоматки

ДОНУШЕН К ЗАЩИТЕ

Заведующий кафедрой РТиТСА
кандидат технических
наук, профессор
Ожикенов К. А.
«29» 06 2025 г.



ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
к дипломному проекту

На тему: «Исследование методов визуализации и анализа для обнаружения трещин в
трубах»

6B07111 – Робототехника и мехатроника

Выполнил

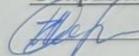
Юсков Даниил

Рецензент

Научный руководитель

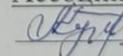
Старший преподаватель, PhD

Кандидат технических наук,

 Алимбаева Ж.Н.

Ассоциированный профессор

«29» май 2025 г.

 Курмангалиева Л.А.

«29» май 2025 г.

Алматы 2025

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ
КАЗАХСТАН

Некоммерческое акционерное общество «Казахский национальный
исследовательский технический университет имени К.И.Сатпаева»



SATBAYEV
UNIVERSITY

Институт Автоматики и информационных технологий

Кафедра Робототехники и технических средств автоматизи

6B07111 – Робототехника и мехатроника

УТВЕРЖДАЮ
Заведующий кафедрой РТиТСА
кандидат технических
наук, профессор
Ожигенов К. А.
«9» 06 2025 г.

ЗАДАНИЕ

на выполнение дипломного проекта

Студенту Юскову Даниилу

1.Тема: Исследование методов визуализации и анализа для обнаружения трещин в трубах

Утверждена приказом ректора _____ № 521-1/18 от «13» 11 2024 г.

2.Срок сдачи законченной работы «30» 05 2025 г.

3.Исходные данные к дипломному проекту:

4.Перечень подлежащих разработке в дипломном проекте вопросов:

1) Исследование, позволяющее понять принцип и перспективы использования машинного и глубокого изучения в задачах детекции трещин

2) Обучение моделей на подготовленных данных, настройка параметров обучения

3) Сравнительный анализ эффективности моделей

5.Перечень графического материала (чертежи; таблицы)

Рисунки: 17

Таблицы: 1

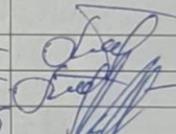
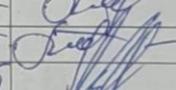
6. Перечень основной рекомендуемой литературы: 11 наименований

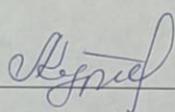
ГРАФИК
подготовки дипломного проекта

Наименование разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю	Примечания
Теоретическая часть	14.03.2025г	50%
Исследовательская часть	15.04.2025г	80%
Практическая часть	12.05.2025г	90%
Итого	30.05.2025г	100%

Подписи

консультантов и норм контролера на законченный дипломный проект с указанием относящихся к ним разделов проекта

Наименование разделов	Консультанты, И.О.Ф. (уч. степень, звание)	Дата подписания	Подпись
Теоретическая часть	Курмангалиева Л.	9.06.2025	
Практическая часть	Курмангалиева Л.Н.	9.06.2025	
Норм контролер	Калимулла Е.У.	09.06.2025	

Научный руководитель:  — Курмангалиева Л. А.

Задание принял к исполнению обучающийся:  Юсков Д.В

Дата « 9 » 06 2025

АНДАТПА

Дипломдық жоба құбырлардағы жарықтарды анықтауға арналған бейнелеу және кескінді талдау әдістерін зерттеуге арналған. Жобаның мақсаты диагностиканың дәлдігін жақсарту үшін ең тиімді алгоритмдер мен тәсілдерді анықтау мақсатында құбырлардағы жарықтарды анықтау үшін қолданылатын бейнелеу және кескінді өңдеу әдістерін зерттеу және салыстырмалы талдау болып табылады.

Теориялық бөлімде жарықтардың пайда болу себептері, олардың түрлері, сондай-ақ визуалды және автоматтандырылған бақылаудың қолданыстағы әдістері қарастырылады. Кескінді өңдеудің заманауи технологияларына шолу жасалады және практикалық бөлімді жүзеге асыру үшін ең қолайлы Алгоритмдер таңдалады.

Практикалық бөлімде терең оқытудың екі моделі: YOLOv8 және DeepLabV3 көмегімен жарықтарды анықтау жүйесі енгізілді және сыналды. Осы модельдерді оқыту және тексеру арқылы олардың дәлдігі мен жарықшақтарды сегменттеу тапсырмасына жарамдылығы салыстырылды. Бұл техникалық бақылау жүйелерінде қолданған кезде әр тәсілдің күшті және әлсіз жақтарын анықтауға мүмкіндік берді.

АННОТАЦИЯ

Дипломный проект посвящен исследованию методов визуализации и анализа изображений для обнаружения трещин в трубах. Целью проекта является исследование и сравнительный анализ методов визуализации и обработки изображений, применяемых для обнаружения трещин в трубопроводах, с целью определения наиболее эффективных алгоритмов и подходов для повышения точности диагностики.

В теоретической части рассматриваются причины появления трещин, их виды, а также существующие методы визуального и автоматизированного контроля. Проводится обзор современных технологий обработки изображений и выбираются наиболее подходящие алгоритмы для реализации практической части.

В практической части была реализована и протестирована система для обнаружения трещин с использованием двух моделей глубокого обучения: YOLOv8 и DeepLabV3. С помощью обучения и проверки этих моделей проведено сравнение их точности и пригодности для задачи сегментации трещин. Это позволило выявить сильные и слабые стороны каждого подхода при использовании в системах технического контроля.

ABSTRACT

The diploma project is devoted to the study of visualization and image analysis methods for crack detection in pipes. The aim of the project is to study and comparative analysis of visualization and image processing methods used to detect cracks in pipelines in order to determine the most effective algorithms and approaches to improve the accuracy of diagnosis.

The theoretical part considers the causes of cracks, their types, as well as existing methods of visual and automated control. Modern image processing technologies are reviewed and the most suitable algorithms are selected for the realization of the practical part.

In the practical part, a system for crack detection was implemented and tested using two deep learning models: YOLOv8 and DeepLabV3. By training and validating these models, their accuracy and suitability for the crack segmentation task was compared. This allowed to identify the strengths and weaknesses of each approach when used in inspection systems.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	8
1. Методы визуализации	9
1.1 Актуальность проблемы трещинообразования в трубопроводах	9
1.2 Опасность трещин и необходимость раннего выявления	11
1.3 Обзор существующих методов диагностики и визуального контроля	12
2. Методика и инструменты исследования	14
2.1 Выбор программных средств и инструментов анализа	14
2.2 Подготовка и описание датасетов	17
3. Практическая реализация и сравнительный анализ	20
3.1 Обучение моделей	20
3.2 Сравнение YOLOv8 и DeepLabV3	24
Заключение	29
Глоссарий	30
Список использованной литературы	31
Приложение А	32
Приложение Б	33

ВВЕДЕНИЕ

Трубопроводы широко применяются в различных отраслях — от водоснабжения и отопления до нефтегазовой и химической промышленности. Надёжность и безопасность этих систем напрямую зависят от своевременного выявления дефектов, особенно таких, как трещины, которые могут привести к утечкам, авариям и серьёзным экономическим потерям. В связи с этим актуальной задачей становится разработка методов, позволяющих эффективно обнаруживать подобные повреждения.

Одним из самых доступных и наглядных способов диагностики является визуальный контроль, который с развитием цифровых технологий получил мощное расширение — теперь можно применять программную обработку изображений для автоматического анализа состояния поверхностей. Такие методы позволяют выявлять мелкие трещины, анализировать форму дефекта и оценивать его размеры без участия человека.

Классические методы диагностики — визуальный осмотр, ультразвуковая дефектоскопия, рентген — требуют специализированного оборудования и высококвалифицированного персонала, что ограничивает их применение в условиях протяжённых или труднодоступных участков. В этих условиях на первый план выходят технологии автоматического анализа изображений, основанные на методах компьютерного зрения и машинного обучения. Такие подходы позволяют выявлять дефекты по цифровым снимкам в реальном времени, автоматизируя контроль и снижая человеческий фактор.

1. Методы визуализации

1.1 Актуальность проблемы трещинообразования в трубопроводах

Трубопровод – это система труб, используемая для транспортировки жидкостей, газов и других сред. Она используется в различных отраслях, таких как нефтегазовая промышленность, химическая промышленность, водоснабжение и многие другие.[1] В современном мире трубопроводы играют ключевую роль в транспортировке различных веществ: нефти, газа, воды, химических растворов и других жидких или газообразных сред. Они используются в коммунальных сетях, промышленности, теплоэнергетике и нефтегазовой отрасли. Несмотря на прочность используемых материалов и совершенствование технологий прокладки, трубопроводы со временем подвержены износу и повреждениям, наиболее опасными из которых являются трещины. Их образование может привести к серьёзным последствиям — от локальных утечек до масштабных аварий и катастроф.

Трещины являются одним из наиболее распространённых дефектов, возникающих в металлических деталях. Особенно часто они появляются в трубах, используемых для транспортировки жидкостей. Основной причиной образования трещин является воздействие на металл механических и термических нагрузок. В результате этого происходит нарушение внутренней структуры материала разделение кристаллов что и приводит к образованию дефекта, обусловленного усталостью материала. В случае с трубами это означает, что во время эксплуатации внутри них возникает турбулентное давление. Оно вызывает незначительные, но регулярные изменения формы труб, что приводит к постепенному ослаблению материала и возникновению трещин. С течением времени эти трещины могут увеличиваться, создавая угрозу целостности трубопровода.

Если речь идёт о нефти или химических растворах — это грозит значительным загрязнением окружающей среды и нарушением экологической безопасности. Трубопроводы могут использоваться для транспортировки различных сред, таких как нефть, газ, вода и другие. Для каждого типа среды необходимо выбирать соответствующий тип трубопровода и материал.

Трубопроводы являются важной составляющей различных отраслей и используются для транспортировки различных сред. Они могут быть изготовлены из различных материалов, таких как сталь, пластик и другие, и имеют свои особенности и применение. Для обеспечения высокой степени надёжности и долговечности трубопроводов необходимо проводить регулярное обслуживание и замену.[1].

Причины образования трещин в трубопроводах разнообразны. Одна из основных — усталостное разрушение, которое возникает под действием многократных циклических нагрузок. Особенно это характерно для труб, через которые проходят жидкости под переменным давлением или при резких температурных колебаниях. Кроме того, большую роль играют коррозионные процессы. Коррозия ослабляет стенки труб, делая их более уязвимыми к растрескиванию даже при незначительных механических воздействиях. Особенно опасна стресс-коррозия — разрушение материала под воздействием

одновременно коррозионной среды и внутреннего напряжения .

Также к образованию трещин могут привести:

- Механические повреждения при монтаже, транспортировке или эксплуатации;
- Ошибки при сварке (неполный провар, пористость, шлаковые включения);
- Технологические дефекты материала, например, наличие микропустот или включений;
- Внешние воздействия, такие как сейсмическая активность, вибрация, осадки грунта и промерзание .

Классифицируют трещины по разным признакам:

- По направлению: продольные, поперечные, кольцевые;
- По глубине: поверхностные, сквозные, подповерхностные;
- По причине возникновения: термические, коррозионные, усталостные, технологические;

По степени опасности: активные (растущие) и пассивные (стабилизированные)

Особую опасность представляют микротрещины, которые невозможно выявить при обычном визуальном осмотре. Они могут долгое время оставаться незамеченными, развиваясь внутри металла и распространяясь под действием нагрузок. Без своевременной диагностики такие дефекты способны привести к внезапному разрушению трубопровода .

Современные условия эксплуатации требуют регулярного и надёжного контроля за состоянием трубопроводов. Однако, традиционные методы, такие как визуальный осмотр или гидравлические испытания, не всегда эффективны для обнаружения внутренних или подповерхностных трещин. Именно поэтому всё более актуальными становятся методы, основанные на анализе изображений, полученных при помощи камер, установленных внутри труб, или других средств неразрушающего контроля .

Применение таких методов позволяет обнаруживать трещины на ранней стадии, оценивать их форму, длину и направление распространения. Это, в свою очередь, позволяет принять меры до того, как произойдёт авария, тем самым повышая надёжность и безопасность всей системы .

Таким образом, изучение причин образования трещин и разработка эффективных методов их обнаружения являются важной задачей в области технической диагностики трубопроводов. Это особенно актуально в современных условиях, когда возраст многих инженерных сетей превышает нормативный срок службы, а аварии могут привести к катастрофическим последствиям как для производства, так и для экологии.

1.2 Опасность трещин в трубопроводе и необходимость раннего выявления

Возникновение трещин в трубопроводах — это не только структурное нарушение, но и потенциальный источник серьёзной техногенной и экологической угрозы. Даже небольшие по размеру дефекты могут привести к нарушению целостности системы, особенно если речь идёт о трубопроводах, по которым транспортируются опасные вещества: газ, нефть, кислоты или другие химически активные среды.

Одним из главных последствий трещинообразования является утечка транспортируемых веществ. Такая утечка может не сразу быть замечена, особенно если дефект развивается изнутри и не даёт явных внешних признаков. В промышленных условиях это грозит не только потерей продукции, но и повышением давления в системе, что усиливает разрушение и может привести к взрывам или возгораниям — особенно в случае с углеводородами [3]. В бытовых или коммунальных сетях это может вызвать аварийные отключения, подтопления, перебои с подачей воды или тепла.

Дополнительно, трещины зачастую становятся пусковым механизмом коррозии. Через микродефекты внутрь металла проникает влага, кислород и агрессивные соединения, ускоряя электрохимические процессы разрушения. В результате ухудшаются механические свойства материала, и трещина начинает расти с большей скоростью. Особенно быстро протекает этот процесс в трубах, подверженных переменным температурным и гидравлическим режимам.

Экологические последствия трещин могут быть катастрофическими. Аварии на трубопроводах с утечкой нефти или газа вызывают загрязнение почвы, отравление водных источников, гибель флоры и фауны. В городских условиях такие повреждения могут также повлечь за собой эвакуацию населения и нарушение функционирования социальной инфраструктуры.

С экономической точки зрения, последствия выражаются в:

- потерях транспортируемого вещества;
- затратах на устранение аварии;
- простоев оборудования;
- штрафах за экологические нарушения;
- нарушении контрактных обязательств и логистических цепочек.

Серьёзность последствий особенно возрастает при позднем выявлении трещин, что делает своевременную диагностику жизненно необходимой. Однако практика показывает, что большая часть дефектов обнаруживается уже после проявления явных признаков разрушения. Это связано с ограниченностью традиционных методов контроля, таких как внешний осмотр или гидравлические испытания. Они не всегда позволяют обнаружить скрытые или подповерхностные трещины, которые, согласно , являются наиболее коварными и опасными.

Современные условия эксплуатации инженерных систем требуют перехода к прогностическому техническому обслуживанию — когда дефекты выявляются не по факту аварии, а в процессе регулярного мониторинга. Именно поэтому всё большую актуальность приобретают методы визуального контроля с элементами цифровой обработки изображений. Эти технологии позволяют анализировать состояние внутренних поверхностей труб с высокой точностью, выявляя трещины по форме, длине.

Кроме того, автоматизация анализа изображений существенно снижает влияние человеческого фактора. Алгоритмы компьютерного зрения позволяют обрабатывать данные в режиме реального времени, классифицируя трещины по степени опасности и давая возможность оперативно принимать решения о ремонте или замене участка трубопровода .

1.3 Обзор существующих методов диагностики и визуального контроля

Один из самых простых и традиционных методов — это визуальный контроль, выполняемый оператором вручную при помощи фонарей, зеркал, эндоскопов и камер. Он позволяет выявлять крупные дефекты и механические повреждения на доступных участках труб, однако сильно ограничен в эффективности при работе с внутренними поверхностями или при наличии отложений . Главный недостаток визуального осмотра — высокая зависимость от человеческого фактора и невозможность оценки скрытых повреждений.



Рисунок 1. Визуальный осмотр

Более современным способом диагностики считается видеоинспекция с использованием специальных камер. Их запускают внутрь труб с помощью гибких кабелей, тележек или роботов, чтобы получить прямую трансляцию состояния внутренней поверхности. Такие системы обычно оснащены подсветкой, стабилизацией изображения, автофокусом и возможностью записи — это позволяет фиксировать трещины, коррозию, вмятины и другие дефекты, даже в труднодоступных местах. Однако, несмотря на техническую оснащённость, оценка полученных материалов по-прежнему зависит от внимательности и опыта оператора, а значит — человеческий фактор остаётся важным ограничением такого подхода. Для более точного анализа состояния материала и выявления внутренних или скрытых трещин применяются методы

неразрушающего контроля (НК). Среди них:

Ультразвуковой контроль (УЗК) — используется для определения толщины стенок и обнаружения подповерхностных дефектов. Ультразвуковая волна отражается от внутренних неоднородностей, позволяя выявить трещины, даже если они не выходят на поверхность. Ультразвуковой метод основан на анализе процесса распространения упругих колебаний в диагностируемом объекте. Ультразвуковые колебания способны отражаться от внутренних неоднородностей среды, что и является основой для данного метода.[3]

Магнитопорошковый метод — эффективен при контроле ферромагнитных материалов. Основан на выявлении магнитных потоков, искажённых трещинами и другими дефектами.



Рисунок 2. Неразрушающий контроль трубопровода магнитопорошкового метода



Рисунок 3. Неразрушающий контроль трубопровода электромагнитным методом



Рисунок 4. Неразрушающий контроль трубопровода методом рентгенографии

Несмотря на эффективность перечисленных способов, они в большинстве своём остаются точечными, трудоёмкими и зависящими от условий доступа и опыта оператора. В условиях растущих требований к безопасности и масштабам трубопроводных систем всё большую актуальность приобретают автоматизированные методы визуального контроля, основанные на компьютерном зрении и алгоритмах машинного обучения. Эти технологии позволяют проводить диагностику быстро, без физического вмешательства и с минимизацией влияния человеческого фактора, что делает их перспективным направлением развития отрасли.

Глава 2. МЕТОДИКА И ИНСТРУМЕНТЫ ИССЛЕДОВАНИЯ

2.1 Выбор программных средств и инструментов анализа

Современные проблемы технического зрения, включая автоматическое нахождение трещин на поверхности трубопроводов, решаются с применением методов машинного обучения — способа, при котором алгоритмы самостоятельно обнаруживают закономерности в сведениях и принимают решения на их базе. Особую эффективность в подобных задачах продемонстрировали методы машинного и глубокого обучения, базирующиеся на применении искусственных нейронных сетей, способных анализировать изображения, извлекать признаки и выделять дефектные области с высокой точностью.

Для реализации этих методов необходимо применение специализированных программных инструментов, которые обеспечивают:

1. Обработку и подготовку изображений;
2. Построение, обучение и валидацию моделей;
3. Визуализацию результатов и оценку метрик;
4. Работу с большими объёмами данных в локальных или облачных средах.

В данной работе применялись программные средства на базе языка Python, предназначенные для реализации и обучения нейросетевых архитектур на двух различных типах датасетов: с полигональной разметкой (YOLO-формат) и с пиксельными масками (PNG-маски). Использование двух моделей

— Ultralytics YOLOv8 и DeepLabV3 — позволило провести комплексное сравнение методов обнаружения трещин по их локализации и точности сегментации. Проект реализован в локальной среде на Python 3.13 с поддержкой GPU (CUDA), что обеспечило ускоренное обучение и возможность обрабатывать большой объём изображений без потерь в производительности.

Применённые инструменты и библиотеки:

1. Ultralytics YOLOv8 — фреймворк для быстрой реализации задач сегментации на базе YOLO. Применён к датасету с координатной разметкой трещин в формате YOLO Segmentation, с обучением по маскам-контурным полигонам.
2. DeepLabV3 (из torchvision.models) — архитектура глубокого обучения для пиксельной семантической сегментации, использовалась с датасетом, где аннотации представлены в виде PNG-масок. Применяется для точного распознавания формы трещин.
3. Matplotlib и Seaborn — использовались для визуального анализа: построения графиков обучения, визуализации потерь, отображения наложенных масок и сравнения метрик.

Обучение моделей проводилось в 100 эпох с использованием оптимизатора Adam, снижением потерь (loss) и визуализацией результатов.

Adam (Adaptive Moment Estimation) — это распространённый алгоритм оптимизации, который активно используется в глубоком обучении (DL) и машинном обучении (ML)[5]. Он предназначен для эффективного обновления весов сети в процессе обучения путем адаптации скорости обучения для каждого параметра в отдельности. В YOLOv8 применялись также встроенные аугментации (HSV, flip, mosaic), в DeepLab — кастомные трансформации через torchvision.transforms

В качестве моделей для автоматического обнаружения трещин в изображениях трубопроводов в данной работе выбраны две нейросетевые архитектуры: YOLOv8[7] и DeepLabV3[8]. Эти модели представляют два разных подхода в компьютерном зрении — детекцию объектов и семантическую сегментацию — и позволяют провести разносторонний анализ применимости глубокого обучения к задаче технического контроля.

YOLOv8 — модель для детекции объектов. YOLO расшифровывается как You Only Look Once. Это широко известная архитектура компьютерного зрения, которая знаменита в том числе своим огромным количеством версий, представляет из себя целую фундаментальную модель, которую можно использовать для классификации, трекинга объектов на видео, задач pose estimation и тд. Все это – в реальном времени.[6] Главная идея YOLO заключается в том, чтобы разбить изображение на сетку и сразу предсказать координаты ограничивающих рамок (bounding boxes) и классы объектов, содержащихся в этих участках.

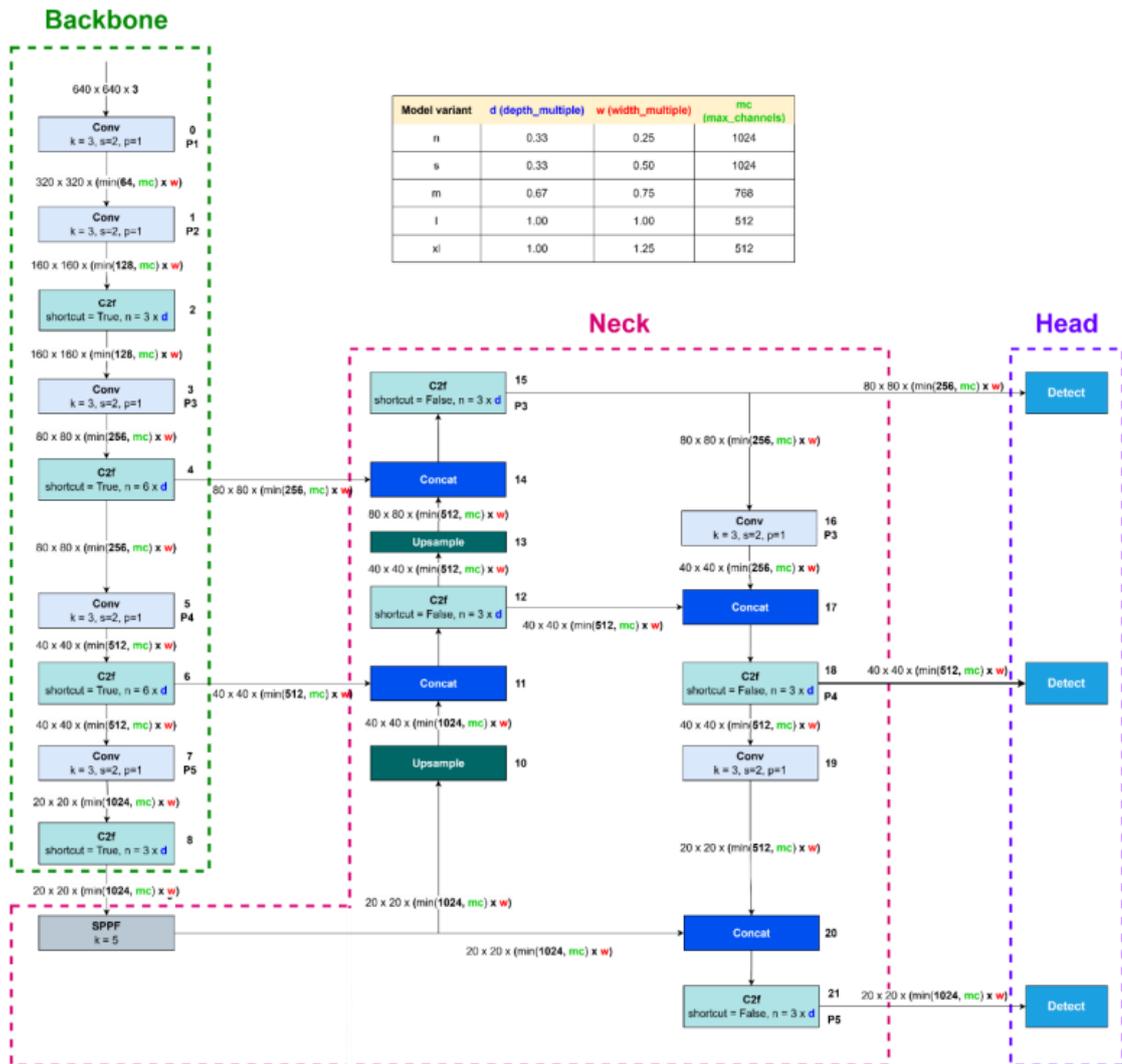


Рисунок 5 Архитектура внутреннего функционала YOLOv8

Основные особенности YOLOv8:

- Одношаговая архитектура: модель одновременно определяет, что присутствует на изображении и где именно.
- Высокая скорость инференса: особенно при использовании GPU, что делает YOLO подходящей для задач в реальном времени.
- Улучшенные блоки C2f и backbone.
- Универсальность: поддержка задач классификации, сегментации и детектирования в рамках единого фреймворка.

DeerLabV3 — модель для семантической сегментации

DeerLabV3 — это одна из ведущих архитектур для семантической сегментации изображений, широко применяемая в задачах, требующих точной локализации объектов на уровне пикселей. В отличие от методов, использующих ограничивающие рамки, данная модель классифицирует каждый пиксель изображения, что особенно важно при анализе дефектов сложной формы, таких как трещины.

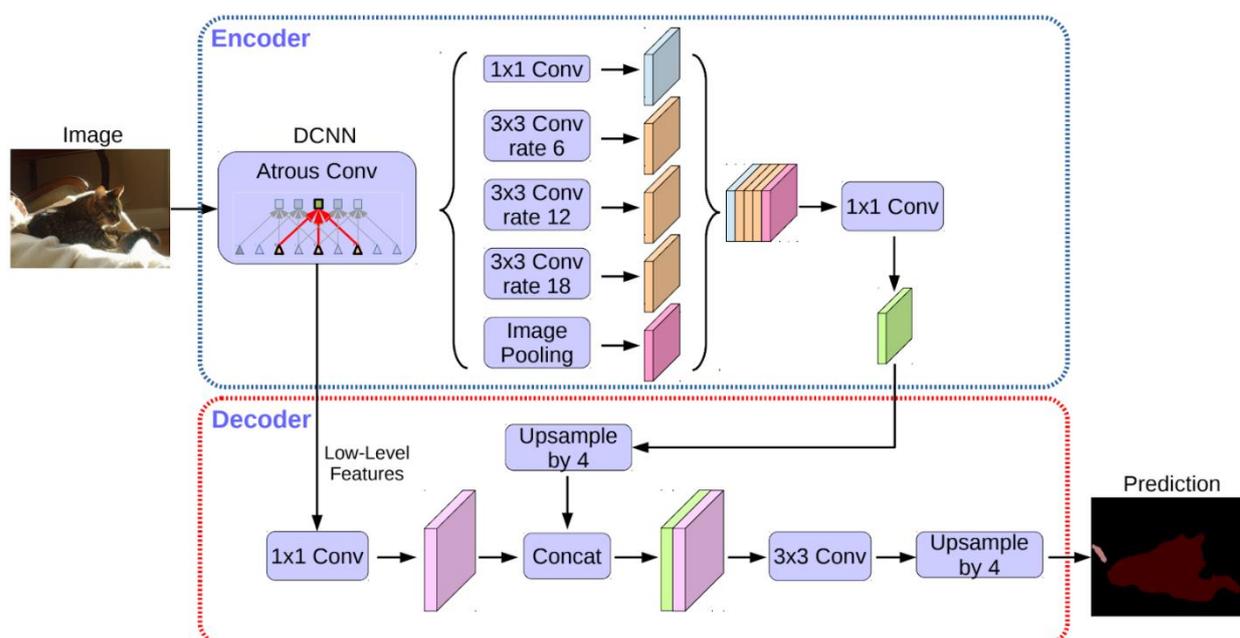


Рисунок 6. Архитектура внутреннего функционала DeepLabV3

Технические особенности DeepLabV3:

- Расширенные свёртки (dilated convolutions): позволяют расширить область восприятия нейрона без потери разрешения;
- Модуль ASPP (Atrous Spatial Pyramid Pooling): извлекает признаки на нескольких масштабах одновременно, что особенно полезно при трещинах разной протяжённости и ориентации;
- Гибкая интеграция с различными скелетами: такими как ResNet или MobileNet, что обеспечивает баланс между точностью и скоростью;
- Пиксельная точность: каждая точка изображения получает метку, что делает модель особенно ценной в задачах промышленной дефектоскопии и контроля качества.

2.2 Подготовка и описание датасетов

В виде исходной информации были использованы общедоступные изображения с трещинами на металлических и бетонных поверхностях труб, а также синтезированные изображения (для имитации условий визуального осмотра внутренней поверхности труб). Главными критериями выбора стали:

- Наличие ясной визуализации трещин;
- Многообразие по размеру, форме и ориентации дефектов;
- Наличие аннотаций или возможность их создания самостоятельно;
- Достаточный объем выборки для обучения и проверки.

Структура первого датасета (YOLOv8)

Этот датасет был адаптирован под формат, совместимый с моделью YOLOv8. Он содержит:

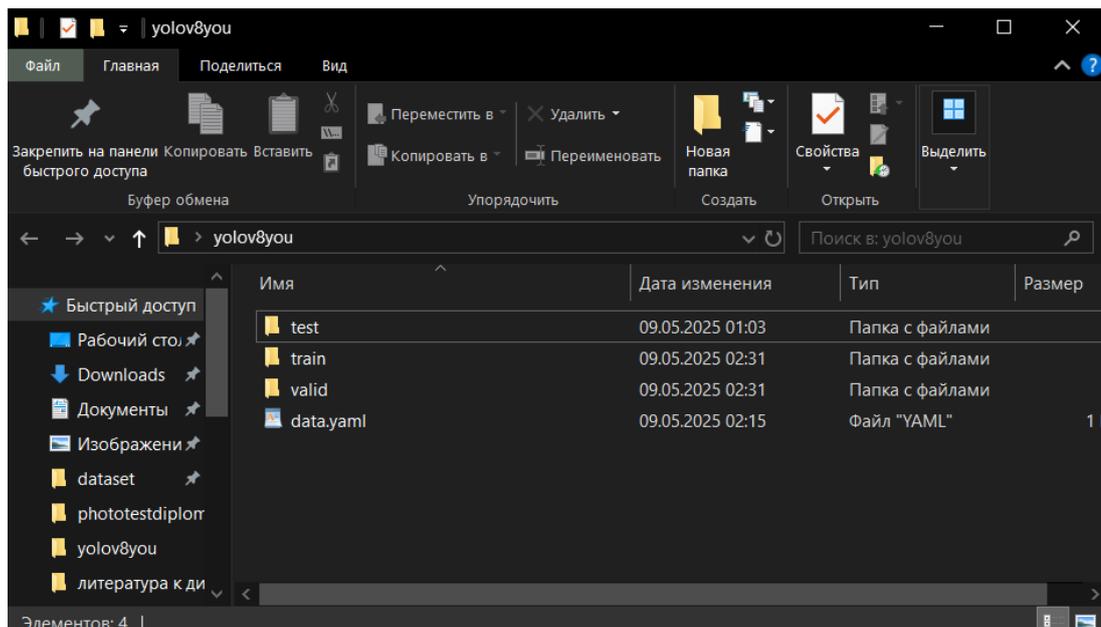


Рисунок 7. Структура набора данных

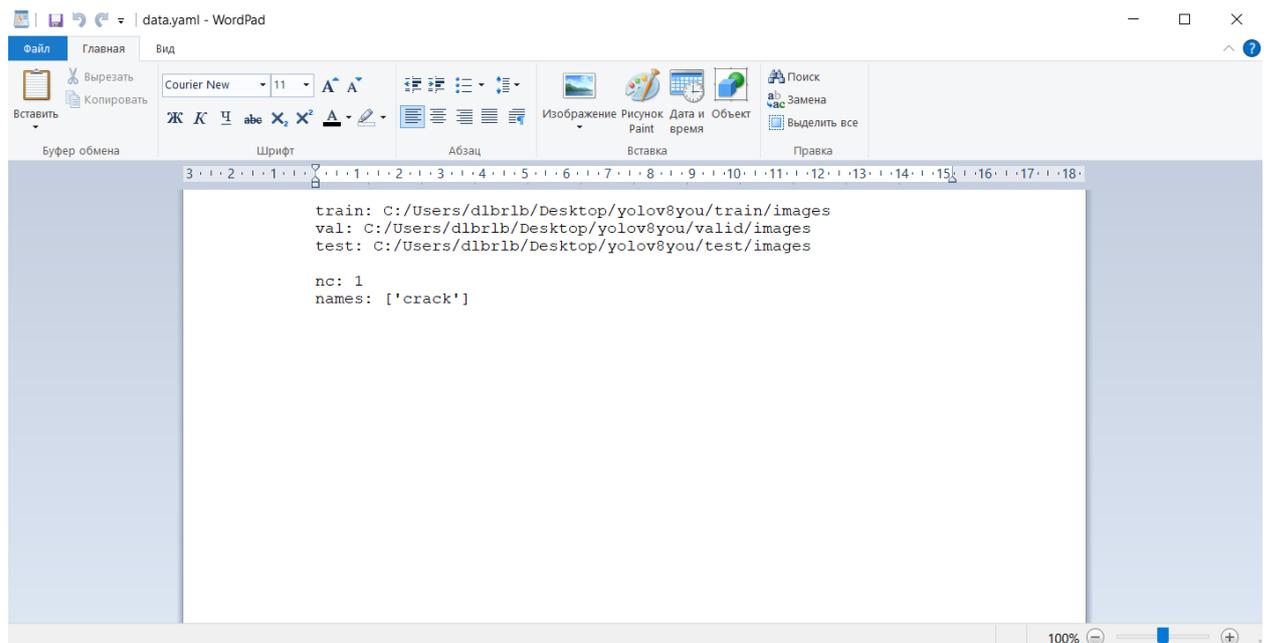


Рисунок 8. Содержимое файла data.yaml для настройки набора данных

Структура набора данных включает в себя:

1. Директория test,train,valid,/ images с исходными изображениями;
2. Директория test,train,valid/labels с аннотациями в формате YOLO (.txt-файлы), Каждая строка аннотации отображает очертания объекта в формате:
class_id x1 y1 x2 y2 x3 y3 ... xn yn
3. Файл конфигурации data.yaml служит для хранения данных о структуре набора данных. В данной работе он применялся для обозначения адресов к изображениям для обучения и валидации. Кроме того, в нём

определялось общее число классов и их названия.

Особенности:

- Координаты нормированы по ширине и высоте изображения;
- Вместо прямоугольников используется контур объекта — масочная сегментация, отражающая форму трещины.

Структура второго датасета (DeerLabV3) содержит:

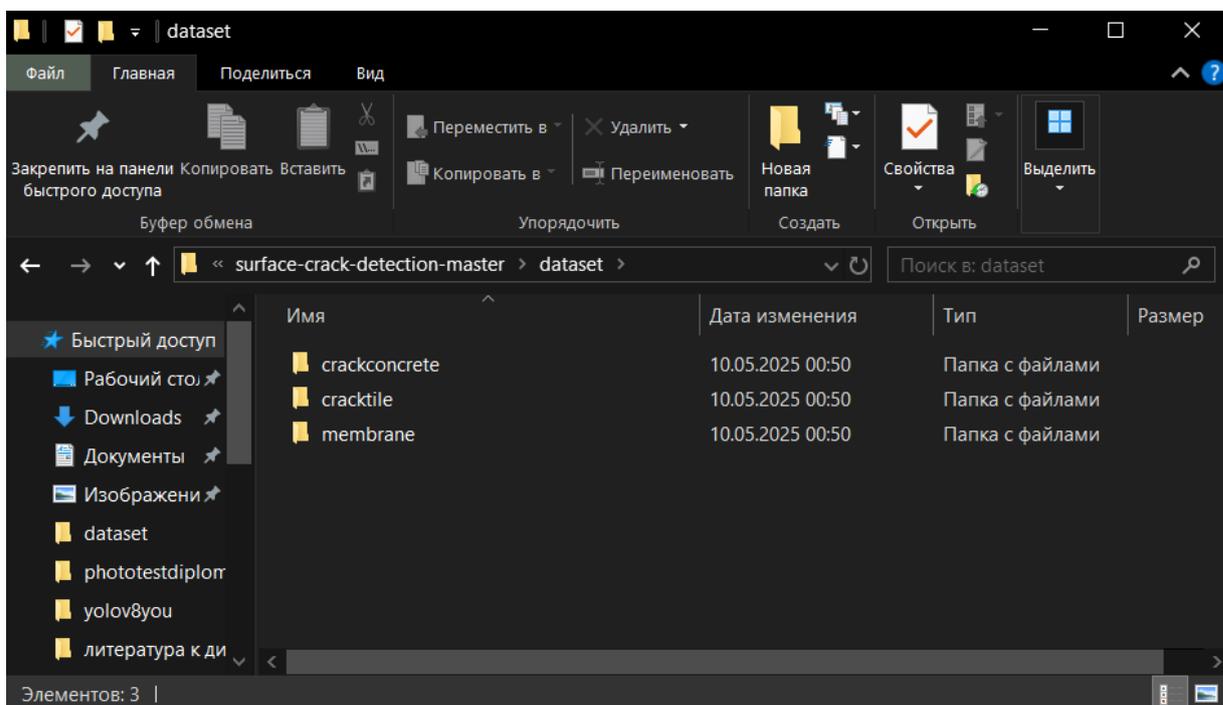


Рисунок 9. Директория набора данных DeerLabV3

Для обучения модели DeerLabV3 применялся датасет с пиксельной сегментацией:

1. Директория cracknocrete , cracktile , membrane/images содержит изначальные изображения;
2. Директория cracknocrete , cracktile , membrane /label — бинарные маски, где трещины - обозначены белым цветом (255), а фон — чёрным (0).

Особенности:

- чёткая граница дефекта на уровне пикселя;
- используется кросс-энтропийная либо Dice loss-функция при обучении.

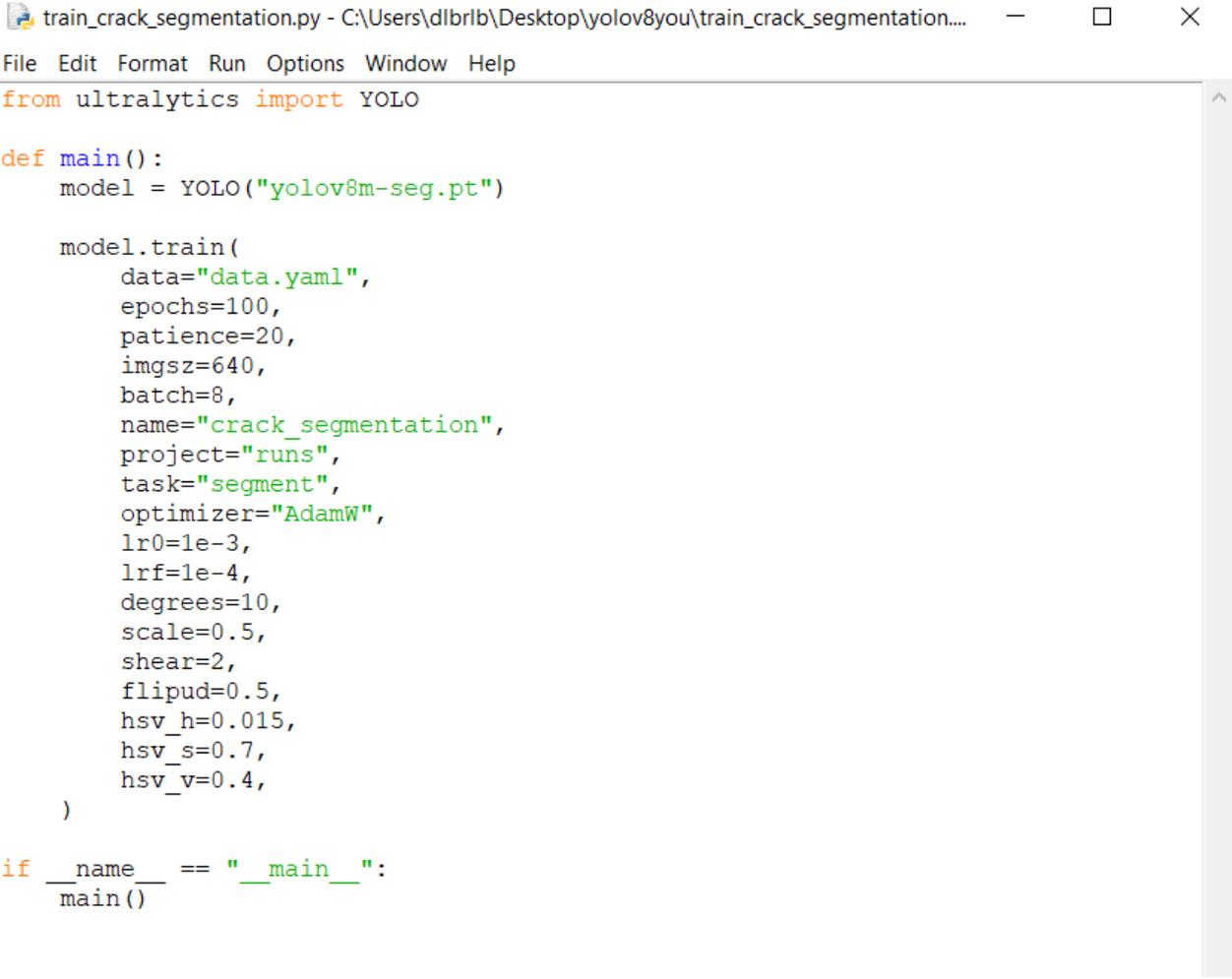
В отличие от YOLO, DeerLabV3 управляется исключительно из Python-кода, что избавляет от необходимости в отдельном файле конфигурации. Вся настройка и параметры задаются непосредственно в коде, минуя этапы создания и редактирования внешних конфигурационных файлов.

Глава 3. ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ И СРАВНИТЕЛЬНЫЙ

АНАЛИЗ

3.1 Обучение моделей

Для старта обучения модели сегментации была применена архитектура YOLOv8m-seg, приспособленная под задачу пиксельной сегментации. Обучение происходит с применением написанного мной Python-скрипта где реализована настройка всех параметров посредством объектной модели Ultralytics.

The image shows a screenshot of a Python script in a code editor. The script is titled 'train_crack_segmentation.py' and is located at 'C:\Users\dlbrlb\Desktop\yolov8you\train_crack_segmentation...'. The code imports the YOLO class from Ultralytics and defines a main function. Inside the main function, a YOLO model is instantiated with 'yolov8m-seg.pt'. The model is then trained with a set of parameters: data='data.yaml', epochs=100, patience=20, imgsz=640, batch=8, name='crack_segmentation', project='runs', task='segment', optimizer='AdamW', lr0=1e-3, lrf=1e-4, degrees=10, scale=0.5, shear=2, flipud=0.5, hsv_h=0.015, hsv_s=0.7, and hsv_v=0.4. The script concludes with a standard if __name__ == '__main__': main() guard.

```
from ultralytics import YOLO

def main():
    model = YOLO("yolov8m-seg.pt")

    model.train(
        data="data.yaml",
        epochs=100,
        patience=20,
        imgsz=640,
        batch=8,
        name="crack_segmentation",
        project="runs",
        task="segment",
        optimizer="AdamW",
        lr0=1e-3,
        lrf=1e-4,
        degrees=10,
        scale=0.5,
        shear=2,
        flipud=0.5,
        hsv_h=0.015,
        hsv_s=0.7,
        hsv_v=0.4,
    )

if __name__ == "__main__":
    main()
```

Рисунок 10. Python-скрипт для обучения модели YOLOv8

Ultralytics — это платформа с открытым исходным кодом, написанная на Python. Её основная задача — обучение и применение моделей YOLO (You Only Look Once), включая модели, способные работать с сегментацией изображений. В рамках данной работы был осуществлён запуск обучения модели YOLOv8-seg. Также была произведена настройка параметров обучения, загрузка набора данных и, как заключительный этап, оценка полученных результатов. Ultralytics предоставляет удобный API высокого уровня, что значительно облегчает интеграцию и управление всем процессом машинного обучения.

Внутри функции main() происходит первоначальная настройка модели. Затем

следует запуск процесса обучения, использующего заранее заданные гиперпараметры.

YOLOv8 автоматически генерирует графики, показывающие распределение объектов, их размеры и положения на изображениях. Эти визуализации помогают быстро оценить качество разметки и выявить ошибки ещё до начала обучения. В DeepLabV3 такой анализ по умолчанию не предусмотрен, что делает YOLO более удобным для предварительной проверки датасета.

Результаты обучения:

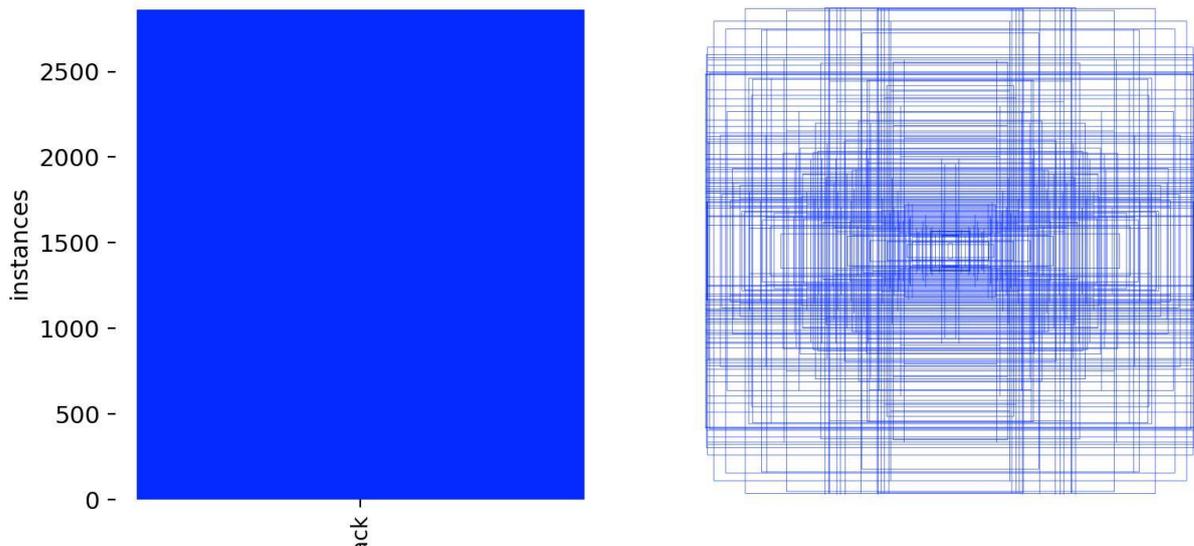


Рисунок 11. Распределение количества трещин в модели YOLOv8

Левый график отображает число экземпляров каждого класса. В данном случае в наборе данных присутствует всего один класс, обозначенный как "crack". Этот класс представлен достаточно равномерно, насчитывая примерно 2500 аннотаций, что свидетельствует об одноклассовой структуре датасета.

На правом графике визуализированы все ограничивающие рамки, наложенные на условное изображение. Наблюдается явное преобладание рамок в центральной области, что указывает на тенденцию возникновения трещин преимущественно в этой зоне изображений. Дополнительно отмечается наличие повторяющихся элементов, которое, вероятнее всего, обусловлено идентичной структурой изображений или схожей локализацией трещин на сгенерированных кадрах.

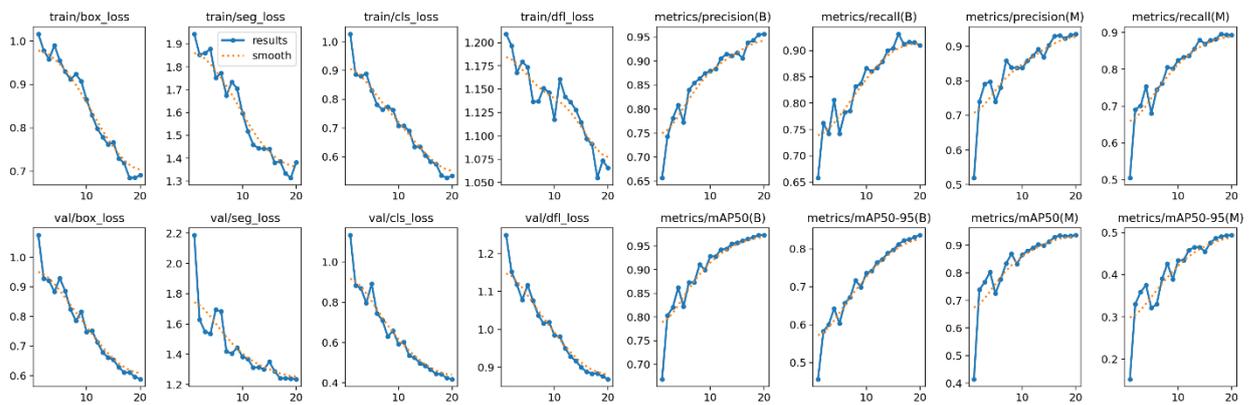


Рисунок 12. Изменение функций качества и метрик потерь при обучении YOLOv8

Проведя сотню итераций обучение модели YOLOv8, Я отслеживал, как меняются ошибки (потери) и точность. На этих графиках показано, как с каждой эпохой ошибки уменьшаются, а точность модели становится лучше.

Результаты тренировки нейронной сети YOLOv8 демонстрируют позитивную и устойчивую тенденцию. На рисунке 12 видно, что все основные параметры потерь, отвечающие за локализацию, сегментацию и классификацию, непрерывно уменьшаются. Параллельно, показатели качества — точность, полнота, средняя точность (mAP) — постоянно увеличиваются, подтверждая эффективность процесса обучения. Отсутствие заметной разницы между результатами свидетельствует об отсутствии переобучения.

Обучение модели на втором датасете DeepLabV3:

```

train_deeplab.py - C:\Users\dibr1b\Desktop\datasetnet_deeplab\surface-crack-detection-master\train_deeplab.py (3.133)
File Edit Format Run Options Window Help
import os
from torch.utils.data import Dataset, DataLoader
from torchvision import transforms
from PIL import Image
import torch
import torch.nn as nn
import torch.optim as optim
from torchvision.models.segmentation import deeplabv3_resnet50
from tqdm import tqdm

NUM_CLASSES = 2
EPOCHS = 100
BATCH_SIZE = 4
LR = 1e-4
DEVICE = 'cuda' if torch.cuda.is_available() else 'cpu'

root_dataset = "C:/Users/dibr1b/Desktop/datasetnet_deeplab/surface-crack-detection-master/dataset"
subdirs = ["crackconcrete", "cracktile", "membrane"]

class CrackSegmentationDataset(Dataset):
    def __init__(self, image_paths, mask_paths, transform=None):
        self.images = image_paths
        self.masks = mask_paths
        self.transform = transform

    def __len__(self):
        return len(self.images)

    def __getitem__(self, idx):
        image = Image.open(self.images[idx]).convert("RGB")
        mask = Image.open(self.masks[idx]).convert("L")

        if self.transform:
            image = self.transform(image)
            mask = self.transform(mask)

        # Маска: 0 - фон, 1 - трещина
        mask = (mask > 0).long().squeeze() # Перевод в классы
        return image, mask

transform = transforms.Compose([
    transforms.Resize((256, 256)),
    transforms.ToTensor()
])

```

Рисунок 13. Python-скрипт для обучения модели DeepLabV3

В ходе работы были задействованы следующие важные библиотеки:
torch, torchvision — основные инструменты для создания, обучения и обработки моделей, а также для работы с данными;

PIL и transforms — инструменты для загрузки изображений и первичной предобработки (например, изменение масштаба и преобразование в тензоры);

tqdm — для визуализации прогресса обучения;

os — для автоматической загрузки изображений и масок, находящихся в подпапках набора данных.

Предварительно данные были разделены на три категории, на основе которых автоматически формировались списки с путями к изображениям и маскам. Обработка изображений и соответствующих масок осуществлялась через пользовательский класс Dataset, подававший данные в модель посредством DataLoader.

Для обучения применялась модель deeplabv3_resnet50, адаптированная к задаче бинарной сегментации путём изменения последнего классификационного слоя.

Функция потерь была выбрана CrossEntropyLoss, поскольку задача представляла собой бинарную классификацию пикселей: трещина или фон. Качество сегментации оценивалось, прежде всего, визуальным соответствием предсказанных масок и исходных 8разметок.

Визуальные результаты работы моделей:

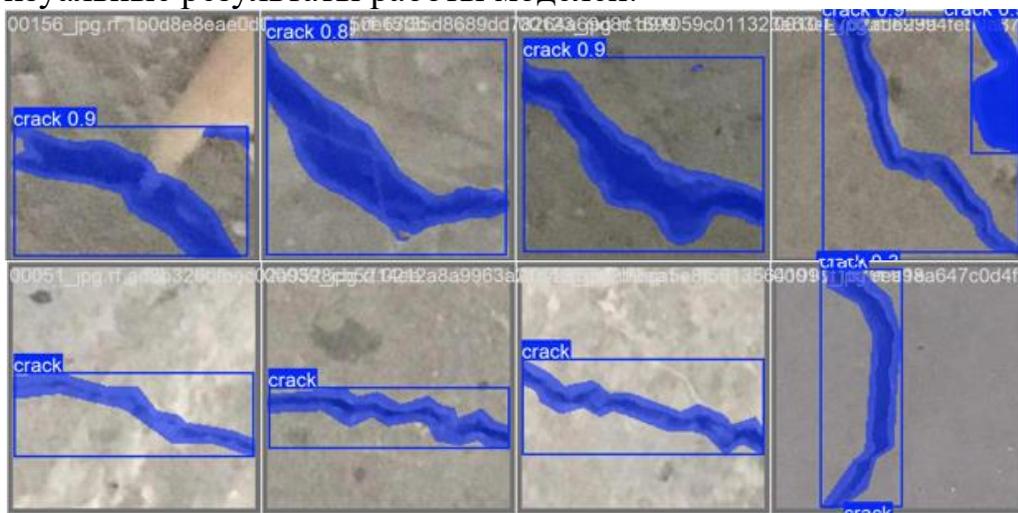


Рисунок 14. Визуально предсказанные маски модели YOLOv8

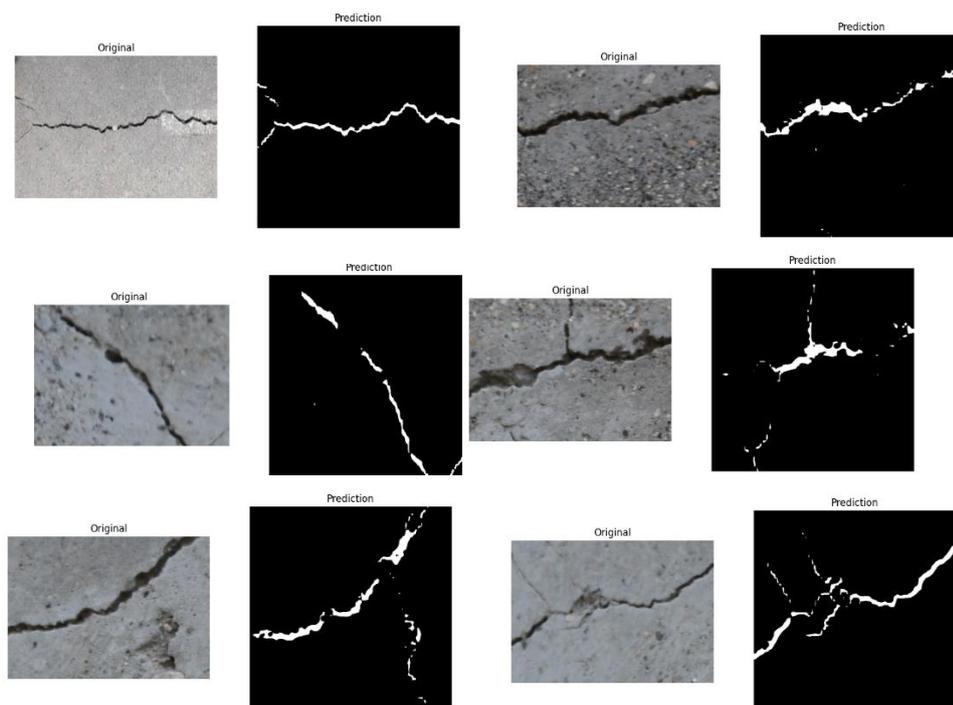


Рисунок 15. Визуально предсказанные маски модели DeepLabv3

Визуально я могу сделать вывод что обученные модели явно имеют недостатки, но все-так же выполняют свою поставленную задачу. Поэтому далее и будет мной проведен расчет метрик для расчёта точности моделей.

3.2 Сравнение YOLOv8 и DeepLabV3

Далее было проведено сравнение моделей. Цель заключалась в том, чтобы оценить, насколько точно каждая из моделей определяет расположение дефектов на изображениях. Сравнение осуществлялось на основе матриц несоответствий (confusion matrix), а также с расчётом ключевых метрик качества precision, recall, IoU, что позволило оценить эффективность алгоритмов.

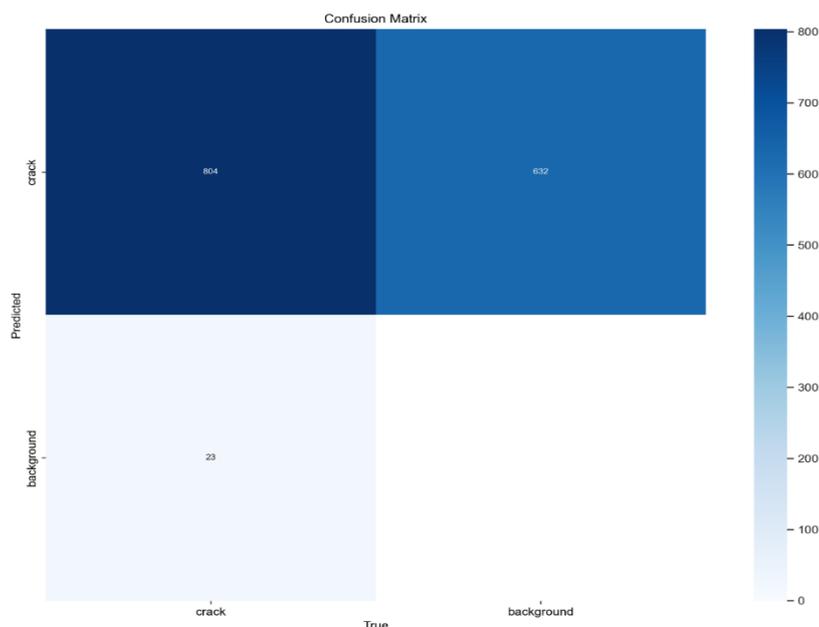


Рисунок 16. Матрица несоответствий модели YOLOv8

804 изображений с трещинами модель правильно распознала (True Positive)

632 изображений без трещин модель ошибочно пометила как трещины (False Positive)

23 изображения с трещинами были пропущены — модель не нашла трещину (False Negative)

TN — True Negative (изображения без трещин, распознаны правильно) не указаны, скорее всего в тестовом наборе все изображения содержали трещины, как и у DeepLab

Расчет метрик. Для того что рассчитать следующие метрики, такие как precision, recall, IoU, F-1 score, я воспользовался уравнением Карла Пирсона[11]

$$\text{Precision(точность)} = \frac{TP}{TP+FP} = \frac{804}{804+632} = \frac{804}{1436} = \frac{804}{1436} \approx 0.5598 \approx 56\% \quad (1)$$

Где:

TP(True Positive) – изображения с правильно распознанными трещинами модели YOLOv8

FP(False Positive) – изображения, на которых модель YOLOv8 при классификации трещин приняла фон за трещины

$$\text{Recall(полнота)} = \frac{TP}{TP+FN} = \frac{804}{804+23} = \frac{804}{827} \approx 0.9722 \approx 97\% \quad (2)$$

TP(True Positive) – изображения с правильно распознанными трещинами модели YOLOv8

FN (False Negative)-изображения на которых модель YOLOv8 не обнаружила трещины

$$\text{IoU} = \frac{TP}{TP+FP+FN} = \frac{804}{804+632+23} = \frac{804}{1459} \approx 0.551 \quad (3)$$

TP(True Positive) – изображения с правильно распознанными трещинами модели YOLOv8

FP(False Positive) – изображения, на которых модель YOLOv8 при классификации трещин приняла фон за трещины

FN (False Negative)-изображения на которых модель YOLOv8 не обнаружила трещины

$$\text{F-1-score} = \frac{2 \times (\text{precision} \times \text{recall})}{\text{precision} + \text{recall}} = \frac{2 \times (0.5598 \times 0.9722)}{0.5598 + 0.9722} = \frac{1.08789}{1.532} \approx 0.7101 \approx 71\% \quad (4)$$

Высокий показатель recall (97%) свидетельствует о том, что YOLOv8 эффективно обнаруживает трещины – она почти никогда их не "пропускает".

Низкий уровень precision (56%) говорит о тенденции модели к избыточной сегментации: она нередко выделяет не только трещины, но и элементы фона.

F1-score (71%) и IoU (55%) демонстрируют, что, хотя модель уверенно идентифицирует дефекты, необходимо улучшить точность, снизив количество ложных срабатываний.

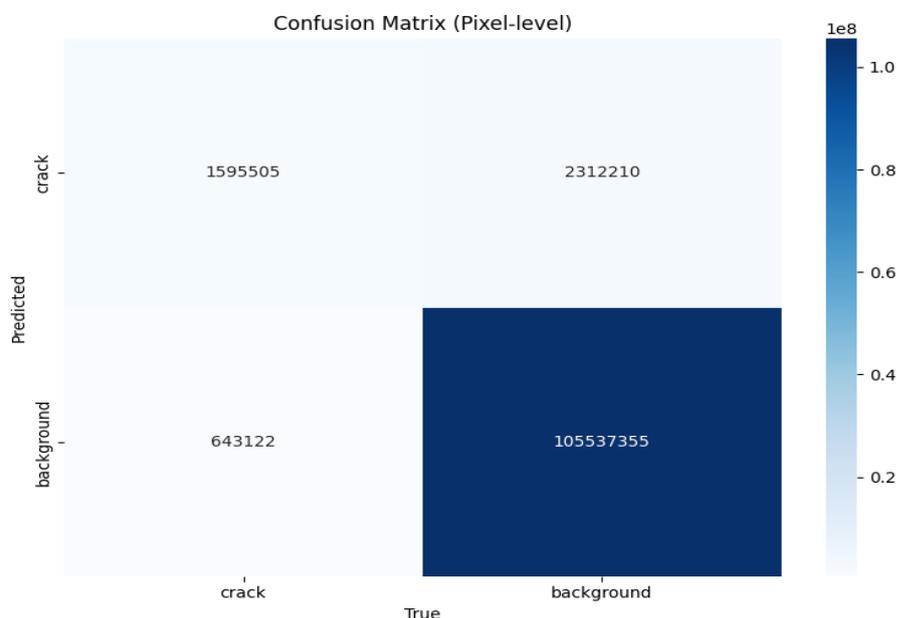


Рисунок 17 Матрица несоответствий модели DeepLabv3

TP=1595505; FP=2312210 ; FN=643122; TN=105537355

Расчет метрик :

$$\text{Precision(точность)} = \frac{TP}{TP+FP} = \frac{1595505}{1595505+2312210} = \frac{1595505}{3907715} \approx 0.408 \approx 41\% \quad (5)$$

$$\text{Recall(полнота)} = \frac{TP}{TP+FN} = \frac{1595505}{1595505+643122} = \frac{1595505}{2238627} \approx 0.7129 \approx 71\% \quad (6)$$

$$\text{IoU} = \frac{TP}{TP+FP+FN} = \frac{1595505}{1595505+2312210+643122} = \frac{1595505}{4550837} \approx 0.337 \quad (7)$$

$$\text{F-1-score} = \frac{2 \times (\text{precision} \times \text{recall})}{\text{precision} + \text{recall}} = \frac{2 \times (0.408 \times 0.7129)}{0.408 + 0.7129} = \frac{0.5817264}{1.1209} \approx 0.4614 \approx 46\% \quad (8)$$

Высокое значение recall, достигающее 71%, свидетельствует о том, что алгоритм DeepLabV3, обнаруживает трещины, однако иногда он всё же не распознаёт их.

Скромный показатель precision, составляющий 41%, указывает на склонность модели к ошибкам: она нередко причисляет к трещинам элементы фона или прочие незначимые области.

Показатели F1-меры (46%) и IoU (33.7%) демонстрируют, что, хотя модель и способна находить дефекты, её точность обводки и соответствие эталону оставляют желать лучшего. Эти данные подразумевают необходимость совершенствования архитектуры, оптимизации пороговых значений либо улучшения баланса классов в процессе обучения.

По итогам сравнения модель YOLOv8 продемонстрировала превосходные результаты по всем основным критериям оценки. Это делает её наиболее подходящим вариантом для автоматизированного обнаружения и сегментации трещин в ситуациях, где критически важно минимизировать пропуск дефектов. В свою очередь, DeepLabV3, хоть и отличается более сложной архитектурой, нуждается в дополнительной настройке. Это касается как обработки фоновых данных, так и оптимизации процесса обучения.

Для обеих моделей — YOLOv8 и DeepLabV3 — я бы рекомендовал использовать аугментацию данных. Аугментация, или генерирование новых данных на основе имеющихся, позволяет довольно просто и дешево решить часть проблем с обучающей выборкой подручными способами. В случае нейронных сетей, распространенным явлением стало встраивать аугментацию непосредственно в процесс обучения, модифицируя данные каждую эпоху[]

Для более наглядного сопоставления я составил сравнительную таблицу, в которой указаны различия между моделями, которые я тестировал и сравнивал YOLOv8 и DeepLabV3. В таблице я показал особенности аннотаций, форматов данных, способов обработки трещин, уровня детализации, а также плюсы и минусы каждой модели. Некоторые пункты основаны на моих личных наблюдениях, полученных в процессе практического тестирования.

Таблица 1. Сравнительные характеристики моделей

Параметр	Датасет для YOLOv8	Датасет для DeepLabV3 (ResNet-50)
Тип задачи	Локализация объектов (обнаружение трещин)	Пиксельная семантическая сегментация (области трещин)
Тип аннотации	Координаты контура трещины (полигон)	Бинарные маски (0 – фон, 255 – трещина)
Формат аннотаций	.txt: список координат образующих полигон	.png или .jpg маски соответствующие каждому изображению
Нормализация координат	Нормированные (0–1)	Абсолютные пиксели
Обработка формы трещин	Точное приближение по координатам (полигон)	Точное описание по пикселям
Уровень детализации аннотаций	Высокий	Высокий
Размер изображений	Разный, масштабируются	Одинаковый (например, 640×640)
Формат изображений	.jpg, .png, .bmp	.png (для сохранения масок)
Совместимость с фреймворками	Ultralytics YOLO	PyTorch, TensorFlow, DeepLabV3
Целевой формат данных	Разметка контуров объектов	Разметка по-пиксельно
Объём аннотированных данных	Высокий (координаты полигона)	Высокий (маски)
Простота подготовки	Средняя	Сложная
Обучаемая модель	YOLOv8	DeepLabV3 (ResNet-50)
Преимущества	Быстрое обучение, высокая скорость инференса	Высокая точность границ, пригодность для технического анализа
Недостатки	Сложность аннотаций, чувствительность к шуму	Высокая трудоёмкость разметки, медленный инференс
Пере или недосегментация	Пересегментация	Недосегментация
Общие выводы	Модель YOLOv8 обеспечивает быстрый анализ и точное приближение форм, проста в использовании	Модель DeepLabV3 сложна, нуждается в большем обучении.

Вывод по главе:

В процессе обучения и анализа моделей сегментации YOLOv8 и DeepLabV3 для задачи детекции трещин были получены следующие результаты:

Проведено обучение модели YOLOv8 с использованием аннотаций и настроек, реализованных через Ultralytics.

Выполнено обучение модели DeepLabV3 на основе бинарных масок с применением библиотеки PyTorch.

Для оценки качества сегментации использовались метрики Карла Пирсона : IoU, , Recall и F1-score.

Проведено сравнительное тестирование моделей по результатам матриц ошибок и метрик качества. На моем примере выявлено, что YOLOv8 показывает более высокую полноту и пригодна для задач, где важно минимизировать пропуски дефектов, тогда как DeepLabV3 требует дополнительной оптимизации для повышения точности.

ЗАКЛЮЧЕНИЕ

В рамках выполнения дипломного проекта проведена исследовательская и деятельность, целью которой стало изучение и сравнительный анализ методик визуализации и обработки изображений для обнаружения трещин в трубопроводах. Основное внимание было уделено применению современных нейросетевых архитектур, таких как YOLOv8 и DeepLabV3, для решения задач автоматического выявления и сегментации дефектов на внутренних поверхностях труб.

На теоретическом этапе были исследованы причины возникновения трещин, их классификация и потенциальные последствия в случае несвоевременного обнаружения. Также был выполнен обзор существующих методов контроля, включая визуальные, неразрушающие и автоматизированные подходы, основанные на технологиях компьютерного зрения.

Практическая часть включала в себя отбор и подготовку нескольких наборов данных с изображениями трещин, а также обучение моделей YOLOv8 и DeepLabV3. Каждая модель была обучена на специально подготовленных данных, предварительно размеченных и прошедших обработку.

В результате проведённого сравнения установлено, что обе модели успешно решают задачу обнаружения трещин, однако демонстрируют различную специфику: YOLOv8 обеспечивает высокую скорость инференса и оптимальна для задач в реальном времени, в то время как DeepLabV3 демонстрирует более точное определение геометрии дефектов благодаря пиксельной сегментации. Также отмечено преимущество YOLOv8 в плане удобства подготовки и анализа аннотаций.

Анализ показал, что применение нейросетевых методов значительно повышает эффективность раннего выявления дефектов в трубопроводах.

В качестве перспектив дальнейшего развития предлагается увеличить объём обучающей выборки, использовать более специализированные архитектуры сегментации, а также интегрировать обученные модели в реальные системы инспекции с использованием роботизированных платформ и камер высокого разрешения.

ГЛОССАРИЙ

НК-неразрушающий контроль

Инференс — это процесс применения уже обученной модели к новым, ранее не виденным данным для получения предсказаний.

YOLO (You Only Look Once) — это алгоритм и библиотека для обнаружения объектов на изображениях и видео в реальном времени.

DeepLab — это семейство алгоритмов для семантической сегментации изображений

Итерация — это однократное выполнение полного цикла вычислений или действий в процессе обучения модели или при работе алгоритма.

Precision- показывает, насколько правильны предсказания модели среди тех случаев, где она "нашла трещины".

Recall- показывает, насколько модель вообще нашла все трещины, которые были на изображении.

IoU (Intersection over Union, пересечение над объединением)-показывает, насколько предсказанная область трещины совпадает с эталонной.

F1-score — это сбалансированная метрика, которая учитывает и Precision, и Recall.

Простыми словами: одна цифра, которая показывает общее качество — насколько модель хорошо и правильно находит трещины и не пропускает их.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

- [1] Трубопровод: типы, применение и особенности https://re-st.ru/interesnoe/truboprovod-tipy-primeneniye-i-osobennosti/?utm_source=chatgpt.com
- [2] PIPE CRACKS DETECTION METHODS – A REVIEW
https://ijomam.com/wp-content/uploads/2023/01/pag.-114-119_PIPE-CRACKS-DETECTION-METHODS.pdf
- [3] ISSN 2072-0297 Молодой учёный(с 185-187) <https://moluch.ru/archive/102/23388/>
- [4] Что такое машинное обучение? <https://aws.amazon.com/ru/what-is/machine-learning/#:~:text=%D0%9C%D0%B0%D1%88%D0%B8%D0%BD%D0%BD%D0%BE%D0%B5%20%D0%BE%D0%B1%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5%20%E2%80%93%D1%8D%D1%82%D0%BE%20%D0%BD%D0%B0%D1%83%D0%BA%D0%B0%20%D0%BE,%D0%BD%D0%B0%20%D1%88%D0%B0%D0%B1%D0%BB%D0%BE%D0%BD%D1%8B%20%D0%B8%20%D0%BB%D0%BE%D0%B3%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B5%20%D0%B2%D1%8B%D0%B2%D0%BE%D0%B4%D1%8B.>
- [5] Адам Оптимизатор <https://www.ultralytics.com/ru/glossary/adam-optimizer>
- [6] История YOLO <https://habr.com/ru/articles/865834/>
- [7] Pipeline Crack Detection Computer Vision Project <https://universe.roboflow.com/masimba/pipeline-crack-detection>
- [8] surface-crack-detection <https://github.com/arthurflor23/surface-crack-detection>
- [9] YOLOv8 to YOLO11: A Comprehensive Architecture In-depth Comparative Review <https://arxiv.org/pdf/2501.13400>
- [10] A Guide to Using DeepLabV3 for Semantic Segmentation <https://www.datature.io/blog/a-guide-to-using-deeplabv3-for-semantic-segmentation>
- [11] Understanding Confusion Matrix <https://communities.sas.com/t5/SAS-Communities-Library/Understanding-Confusion-Matrix/ta-p/614884>

ПРИЛОЖЕНИЕ А

```
from ultralytics import YOLO

def main():
    model = YOLO("yolov8m-seg.pt")

    model.train(
        data="data.yaml",
        epochs=100,
        patience=20,
        imgsz=640,
        batch=8,
        name="crack_segmentation",
        project="runs",
        task="segment",
        optimizer="AdamW",
        lr0=1e-3,
        lrf=1e-4,
        degrees=10,
        scale=0.5,
        shear=2,
        flipud=0.5,
        hsv_h=0.015,
        hsv_s=0.7,
        hsv_v=0.4,
    )

if __name__ == "__main__":
    main()
```

ПРИЛОЖЕНИЕ Б

```
import os
from torch.utils.data import Dataset, DataLoader
from torchvision import transforms
from PIL import Image
import torch
import torch.nn as nn
import torch.optim as optim
from torchvision.models.segmentation import deeplabv3_resnet50
from tqdm import tqdm

# Конфигурация
NUM_CLASSES = 2
EPOCHS = 100
BATCH_SIZE = 4
LR = 1e-4
DEVICE = 'cuda' if torch.cuda.is_available() else 'cpu'

# Путь к основной папке
root_dataset = "C:/Users/dlbrlb/Desktop/datasetynet_deeplab/surface-crack-detection-master/dataset"
subdirs = ["crackconcrete", "cracktile", "membrane"]

# Кастомный Dataset
class CrackSegmentationDataset(Dataset):
    def __init__(self, image_paths, mask_paths, transform=None):
        self.images = image_paths
        self.masks = mask_paths
        self.transform = transform

    def __len__(self):
        return len(self.images)

    def __getitem__(self, idx):
        image = Image.open(self.images[idx]).convert("RGB")
        mask = Image.open(self.masks[idx]).convert("L")

        if self.transform:
            image = self.transform(image)
            mask = self.transform(mask)

        # Маска: 0 — фон, 1 — трещина
        mask = (mask > 0).long().squeeze() # Перевод в классы
        return image, mask

# Трансформации
transform = transforms.Compose([
    transforms.Resize((256, 256)),
    transforms.ToTensor()
])
```

```

# Сбор путей
image_paths, mask_paths = [], []
for sub in subdirs:
    image_dir = os.path.join(root_dataset, sub, "train", "image")
    label_dir = os.path.join(root_dataset, sub, "train", "label")

    if os.path.exists(image_dir) and os.path.exists(label_dir):
        images = sorted(os.listdir(image_dir))
        masks = sorted(os.listdir(label_dir))

        for img, msk in zip(images, masks):
            image_paths.append(os.path.join(image_dir, img))
            mask_paths.append(os.path.join(label_dir, msk))
    else:
        print(f"Не найдены директории: {image_dir} или {label_dir}")

# Dataset и DataLoader
dataset = CrackSegmentationDataset(image_paths, mask_paths, transform)
dataloader = DataLoader(dataset, batch_size=BATCH_SIZE, shuffle=True)

# DeepLabV3 модель
model = deeplabv3_resnet50(pretrained=True)
model.classifier[4] = nn.Conv2d(256, NUM_CLASSES, kernel_size=1)
model = model.to(DEVICE)

# Оптимизатор и функция потерь
optimizer = optim.Adam(model.parameters(), lr=LR)
criterion = nn.CrossEntropyLoss()

# Обучение
for epoch in range(EPOCHS):
    model.train()
    epoch_loss = 0
    loop = tqdm(dataloader, leave=False)
    for images, masks in loop:
        images, masks = images.to(DEVICE), masks.to(DEVICE)

        optimizer.zero_grad()
        outputs = model(images)['out']
        loss = criterion(outputs, masks)
        loss.backward()
        optimizer.step()

    epoch_loss += loss.item()
    loop.set_description(f"Epoch [{epoch+1}/{EPOCHS}]")
    loop.set_postfix(loss=loss.item())

    print(f" Epoch {epoch+1}/{EPOCHS} — Loss: {epoch_loss:.4f}")

# Сохраняем обученную модель
torch.save(

```

РЕЦЕНЗИЯ

на дипломный проект студента образовательной программы
6В07111 – «Робототехника и мехатроника»
Юскова Даниила Владимировича

Тема работы: «Исследование методов визуализации и анализа для обнаружения трещин в трубах»

Объём выполненной работы:

- а) графическая часть — 14 листов
- б) пояснительная записка — 24 страниц

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Дипломный проект Юскова Даниила посвящена актуальной и практически значимой задаче повышения эффективности контроля состояния трубопроводных систем за счёт внедрения современных технологий компьютерного зрения. В работе рассмотрены и сравнительно проанализированы два подхода к автоматическому обнаружению дефектов — модель детекции объектов YOLOv8 и модель семантической сегментации DeepLabV3.

Структура работы логична и последовательна. Теоретическая часть содержит обоснование актуальности выбранной темы, обзор факторов, влияющих на образование трещин в трубах, и анализ современных методов диагностики. Практическая часть включает описание этапов разработки, подготовки данных, реализации и обучения моделей, а также анализ качества полученных результатов с применением обоснованных метрик (Precision, Recall, IoU, F1-score). Представлены визуализации, иллюстрирующие поведение моделей на тестовых данных.

Особо стоит отметить самостоятельность автора в реализации программной части проекта, его уверенное владение современными инструментами и фреймворками для машинного обучения на языке Python, а также грамотную интерпретацию полученных результатов.

НЕДОСТАТКИ И ЗАМЕЧАНИЯ

В целях повышения научной строгости работы целесообразно было бы уделить больше внимания обсуждению возможных ограничений применяемых моделей и их устойчивости к внешним факторам, а также детализировать возможные пути интеграции предложенных решений в реальные производственные процессы. Кроме того, в ряде фрагментов пояснительной записки стиль изложения можно было бы сделать более академически выдержанным.

Тем не менее указанные замечания не снижают общей высокой оценки проделанной работы.

ЗАКЛЮЧЕНИЕ

Представленная работа свидетельствует о высоком уровне теоретической подготовки студента, его способности к самостоятельному решению инженерных задач с использованием современных методов искусственного интеллекта, а также об умении грамотно обосновывать и анализировать результаты. Работа выполнена на высоком научно-техническом уровне и заслуживает оценки «**отлично**» с допуском к защите.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ КАЗАХСТАН
КАЗАХСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им.
К.И. САТПАЕВА

Рецензент



Старший преподаватель кафедры информационных технологий и библиотечного дела Казахского национального женского педагогического университета, PhD Алимбаева Ж. Н.

« 29 » 05 2025 ж

ОТЗЫВ
НАУЧНОГО РУКОВОДИТЕЛЯ

на дипломный проект

(наименования вида работы)

Юсков Даниил Владимирович

(Ф.И.О. обучающегося)

6B07111 - "Робототехника и мехатроника"

(название специальности и шифр)

Тема: «Исследование методов визуализации и анализа для обнаружения трещин в трубах»

Дипломный проект посвящен исследованию методов визуализации и анализа изображений для обнаружения трещин в трубах. Целью работы является исследование и сравнительный анализ методов визуализации и обработки изображений, применяемых для обнаружения трещин в трубопроводах, с целью определения наиболее эффективных алгоритмов и подходов для повышения точности диагностики.

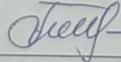
В теоретической части рассматриваются причины появления трещин, их виды, а также существующие методы визуального и автоматизированного контроля. Проводится обзор современных технологий обработки изображений и выбираются наиболее подходящие алгоритмы для реализации практической части.

В практической части была реализована и протестирована система для обнаружения трещин с использованием двух моделей глубокого обучения: YOLOv8 и DeepLabV3.С помощью обучения и проверки этих моделей проведено сравнение их точности и пригодности для задачи сегментации трещин. Это позволило выявить сильные и слабые стороны каждого подхода при использовании в системах технического контроля.

Задачи, поставленные для данного дипломного проекта, были выполнены, студент Юсков Д. продемонстрировал свои навыки и знания при выполнении дипломного проекта, поэтому считаю, что он заслуживает академической степени бакалавра робототехники и мехатроники, и оцениваю работу в 90 баллов.

Научный руководитель

К.т.н., ассоциированный профессор



Л.А. Курмангалиева

9.06

2025 г.



Отчет подобия

Метаданные

Название организации

Satbayev University

Название

Исследование метод визуализации и анализа изображений для обнаружения трещин в трубах

Автор

Научный руководитель / Эксперт

Юсков Даниил Владимирович/Ляззат Курмангалиева

Подразделение

ИАИИТ

Объем найденных подоби

КП-ия определяют, какой процент текста по отношению к общему объему текста был найден в различных источниках. Обратите внимание! Высокие значения коэффициентов не означают плагиат. Отчет должен быть проанализирован экспертом.



КП1

25

Длина фразы для коэффициента подобия 2



КП2

5048

Количество слов



KC

41177

Количество символов

Тревога

В этом разделе вы найдете информацию, касающуюся текстовых искажений. Эти искажения в тексте могут говорить о ВОЗМОЖНЫХ манипуляциях в тексте. Искажения в тексте могут носить преднамеренный характер, но чаще, характер технических ошибок при конвертации документа и его сохранении, поэтому мы рекомендуем вам подходить к анализу этого модуля со всей долей ответственности. В случае возникновения вопросов, просим обращаться в нашу службу поддержки.

Замена букв		6
Интервалы		0
Микропробелы		94
Белые знаки		0
Парафразы (SmartMarks)		5

Подобия по списку источников

Ниже представлен список источников. В этом списке представлены источники из различных баз данных. Цвет текста означает в каком источнике он был найден. Эти источники и значения Коэффициента Подобия не отражают прямого плагиата. Необходимо открыть каждый источник и проанализировать содержание и правильность оформления источника.

10 самых длинных фраз		Цвет текста
ПОРЯДКОВЫЙ НОМЕР	НАЗВАНИЕ И АДРЕС ИСТОЧНИКА URL (НАЗВАНИЕ БАЗЫ)	КОЛИЧЕСТВО ИДЕНТИЧНЫХ СЛОВ (ФРАГМЕНТОВ)
1	https://slidetodoc.com/1-https-ru-m-wikipedia-orgwiki-2-https/	33 0.65 %
2	https://habr.com/ru/articles/865834/	23 0.46 %
3	https://habr.com/ru/articles/865834/	22 0.44 %
4	http://present5.com/mass-media-work-done-by-igor-ponomaryov/	14 0.28 %

5	Сарсеналиев Даурен Оценка качества работы интеллектуальных сетей в информационных системах (1).docx 6/8/2023 International IT University (Радиотехника, электроника и телекоммуникации)	12 0.24 %
6	https://isidatadoc.com/http-waqrkin-dc-usindex-qbc-dwaibacarsesvkwonimdaeyzabpoydarevya942681-aaech-http/	12 0.24 %
7	Подбор оборудования для эксплуатации скважин электроцентробежным насосом 12/12/2023 Atyrau University of Oil and Gas n.a. Saif Utebaev (Департамент науки)	11 0.22 %
8	Диссертация (Айбек Кабышев).docx 6/12/2023 Satbayev University (ИАиИТ)	11 0.22 %
9	http://present5.com/severo-zapadnyj-federalnyj-okrug-zykov-a-s/	11 0.22 %
10	https://sktstj.org/1-56779.html	10 0.20 %
из базы данных RefBooks (0.00 %)		
ПОРЯДКОВЫЙ НОМЕР	НАЗВАНИЕ	КОЛИЧЕСТВО ИДЕНТИЧНЫХ СЛОВ (ФРАГМЕНТОВ)
из домашней базы данных (0.85 %)		
ПОРЯДКОВЫЙ НОМЕР	НАЗВАНИЕ	КОЛИЧЕСТВО ИДЕНТИЧНЫХ СЛОВ (ФРАГМЕНТОВ)
1	Диссертация (Айбек Кабышев).docx 6/12/2023 Satbayev University (ИАиИТ)	25 (3) 0.50 %
2	диплом Михалинова 6/6/2024 Satbayev University (Школа транспортной инженерии и логистики)	18 (2) 0.36 %
из программы обмена базами данных (0.46 %)		
ПОРЯДКОВЫЙ НОМЕР	НАЗВАНИЕ	КОЛИЧЕСТВО ИДЕНТИЧНЫХ СЛОВ (ФРАГМЕНТОВ)
1	Сарсеналиев Даурен Оценка качества работы интеллектуальных сетей в информационных системах (1).docx 6/8/2023 International IT University (Радиотехника, электроника и телекоммуникации)	12 (1) 0.24 %
2	Подбор оборудования для эксплуатации скважин электроцентробежным насосом 12/12/2023 Atyrau University of Oil and Gas n.a. Saif Utebaev (Департамент науки)	11 (1) 0.22 %
из интернета (4.34 %)		
ПОРЯДКОВЫЙ НОМЕР	ИСТОЧНИК URL	КОЛИЧЕСТВО ИДЕНТИЧНЫХ СЛОВ (ФРАГМЕНТОВ)
1	https://habr.com/ru/articles/885834/	45 (2) 0.89 %
2	https://present5.com/istoricheskaya-realnost-vjgoax-serii-vypolni-satlin-dmitri/	45 (8) 0.89 %
3	https://isidatadoc.com/1-https-nu-m-wikipedia-sepwiki-2-https/	38 (2) 0.75 %
4	http://present5.com/severo-zapadnyj-federalnyj-okrug-zykov-a-s/	22 (3) 0.44 %

5	http://present5.com/mass-media-work-done-by-igor-omomayov/	19 (2) 0.38 %
6	https://docs.edpsymposium.com/papers/novel-sensor-projects/ai-on-line-inspection-mmwave-radar	13 (2) 0.26 %
7	https://alidetodoc.com/1/1n-warbin-dq-uaindex-qbq-dwa@qanensulewvnmfmdareyazhpozndevyjs04261-search-https/	12 (1) 0.24 %
8	https://ektai.org/1-56779.html	10 (1) 0.20 %
9	http://present5.com/kostyva-karova-tatyana-vladimirovna-moi-soob-24/	9 (1) 0.18 %
10	https://present5.com/moya-semya-u-gody-velkoj-stachestvennoj-vojny-rodgotyvala/	6 (1) 0.12 %

Список принятых фрагментов

ПОРЯДКОВЫЙ НОМЕР	СОДЕРЖАНИЕ	КОЛИЧЕСТВО ИДЕНТИЧНЫХ СЛОВ (ФРАГМЕНТОВ)
	https://present5.com/istoricheskaya-realnost-v-l...	45 (0.89%)
	https://alidetodoc.com/1-https-ru-m-wikipedia-or...	38 (0.75%)
	http://present5.com/severo-zapadnyj-federalnyj-a...	22 (0.44%)
	http://present5.com/mass-media-work-done-by-igor...	19 (0.38%)

Тема: Исследование методов визуализации и анализа изображений для обнаружения трещин в трубах

АННОТАЦИЯ

Дипломдық жұмыс құбырлардағы жарықтарды анықтау үшін кескіңдерді бейнелеу және талдау әдістерін зерттеуге арналған. Жұмыстың мақсаты диагностиканың дәлдігін жақсарту үшін ең тиімді алгоритмдер мен тәсілдерді анықтау мақсатында құбырлардағы жарықтарды анықтау үшін қолданылатын бейнелеу және кескіңді өңдеу әдістерін зерттеу және салыстырмалы талдау болып табылады.

Теориялық бөлімде жарықтардың пайда болу себептері, олардың түрлері, сондай-ақ визуалды және автоматтандырылған бақылаудың қолданыстаны әдістері қарастырылады. Кескіңді өңдеудің заманауи технологияларына шолу жасалады және практикалық бөлімді жүзеге асыру үшін ең қолайлы Алгоритмдер таңдалады.

Практикалық бөлімде терең оқығудың екі моделі: YOLOv8 және DeepLabV3 көмегімен жарықтарды анықтау жүйесі енгізілді және сыналды. Осы модельдерді оқыту және тексеру арқылы олардың дәлдігі мен жарықтарды сегменттеу тапсырмасына жарамдылығы салыстырылды. Бұл теориялық бақылау жүйелерінде қолданған кезде әр тәсілдің күшті және әлсіз жақтарын анықтауға мүмкіндік берді.

АННОТАЦИЯ

Дипломная работа посвящена исследованию методов визуализации и анализа изображений для обнаружения трещин в трубах. Целью работы является исследование и сравнительный анализ методов визуализации и обработки изображений, применяемых для обнаружения трещин в трубопроводах, с целью определения наиболее эффективных алгоритмов и подходов для повышения точности диагностики.

В теоретической части рассматриваются причины появления трещин, их виды, а также существующие методы визуального и автоматизированного контроля. Проводится обзор современных технологий обработки изображений и выбираются наиболее подходящие алгоритмы для реализации практической части.

В практической части была реализована и протестирована система для обнаружения трещин с использованием двух моделей глубокого обучения: YOLOv8 и DeepLabV3. С помощью обучения и проверки этих моделей проведено сравнение их точности и пригодности для задачи сегментации трещин. Это позволило выявить сильные и слабые стороны каждого подхода при использовании в системах технического контроля.

ABSTRACT

This thesis is devoted to the study of image visualization and analysis methods for detecting cracks in pipes. The aim of this work is to investigate and compare image visualization and processing methods used to detect cracks in pipelines in order to determine the most effective algorithms and approaches for improving the accuracy and reliability of diagnostics.

The theoretical part examines the causes of cracks, their types, and existing methods of visual and automated control. A review of modern image processing technologies is conducted, and the most suitable algorithms for the practical part are selected.

In the practical part, a crack detection system was implemented and tested using two deep learning models: YOLOv8 and DeepLabV3. By training and testing these models, their accuracy and suitability for crack segmentation were compared. This allowed us to identify the strengths and weaknesses of each approach when used in technical control systems.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	8
1. Методы визуализации	9
1.1 Актуальность проблемы трещинообразования в трубопроводах	9
1.2 Опасность трещин и необходимость раннего выявления	11
1.3 Обзор существующих методов диагностики и визуального контроля	12
2. Методика и инструменты исследования	14
2.1 Выбор программных средств и инструментов анализа	14
2.2 Подготовка и описание датасетов	17
3. Практическая реализация и сравнительный анализ	20
3.1 Обучение моделей	20
3.2 Сравнение YOLOv8 и DeepLabV3	24
Глоссарий	28
Список использованной литературы	30
Приложение А	31
Приложение Б	32

ВВЕДЕНИЕ

Трубопроводы широко применяются в различных отраслях - от водоснабжения и отопления до нефтегазовой и химической промышленности. Надёжность и безопасность этих систем напрямую зависят от своевременного выявления дефектов, особенно таких, как трещины, которые могут привести к утечкам, авариям и серьёзным экономическим потерям. В связи с этим актуальной задачей становится разработка методов, позволяющих эффективно обнаруживать подобные повреждения.

Одним из самых доступных и наглядных способов диагностики является визуальный контроль, который с развитием цифровых технологий получил мощное расширение - теперь можно применять программную обработку изображений для автоматического анализа состояния поверхностей. Такие методы позволяют выявлять мелкие трещины, анализировать форму дефекта и оценивать его размеры без участия человека.

Классические методы диагностики - визуальный осмотр, ультразвуковая дефектоскопия, рентген - требуют специализированного оборудования и высококвалифицированного персонала, что ограничивает их применение в условиях протяжённых или труднодоступных участков. В этих условиях на первый план выходят технологии автоматического анализа изображений, основанные на методах компьютерного зрения и машинного обучения. Такие подходы позволяют выявлять дефекты по цифровым снимкам в реальном времени, автоматизируя контроль и снижая человеческий фактор

1. Методы визуализации

1.1 Актуальность проблемы трещинообразования в трубопроводах

Трубопровод - это система труб, используемая для транспортировки жидкостей, газов и других сред. Она используется в различных отраслях, таких как нефтегазовая промышленность, химическая промышленность, водоснабжение и многие другие [1]. В современном мире трубопроводы играют ключевую роль в транспортировке различных веществ: нефти, газа, воды, химических растворов и других жидких или газообразных сред. Они используются в коммунальных сетях, промышленности, теплоэнергетике и нефтегазовой отрасли. Несмотря на прочность используемых материалов и совершенствование технологий прокладки, трубопроводы со временем подвержены износу и повреждениям, наиболее опасными из которых являются трещины. Их образование может привести к серьёзным последствиям - от локальных утечек до масштабных аварий и катастроф.

Трещины являются одним из наиболее распространённых дефектов, возникающих в металлических деталях. Особенно часто они появляются в трубах, используемых для транспортировки жидкостей. Основной причиной образования трещин является воздействие на металл механических и термических нагрузок. В результате этого происходит нарушение внутренней структуры материала, разделение кристаллов, что и приводит к образованию дефекта, обусловленного усталостью материала. В случае с трубами это означает, что во время эксплуатации внутри них возникает турбулентное давление. Оно вызывает незначительные, но регулярные изменения формы труб, что приводит к постепенному ослаблению материала и возникновению трещин. С течением времени эти трещины могут увеличиваться, создавая угрозу целостности трубопровода.

Если речь идёт о нефти или химических растворах - это грозит значительным загрязнением окружающей среды и нарушением экологической безопасности. Трубопроводы могут использоваться для транспортировки различных сред, таких как нефть, газ, вода и другие. Для каждого типа среды необходимо выбирать соответствующий тип трубопровода и материал.

Трубопроводы являются важной составляющей различных отраслей и используются для транспортировки различных сред. Они могут быть изготовлены из различных материалов, таких как сталь, пластик и другие, и имеют свои особенности и применения. Для обеспечения высокой степени надёжности и долговечности трубопроводов необходимо проводить регулярное обслуживание и замену [1].

Причины образования трещин в трубопроводах разнообразны. Одна из основных - усталостное разрушение, которое возникает под действием многократных циклических нагрузок. Особенно это характерно для труб, через которые проходят жидкости под переменным давлением или при резких температурных колебаниях. Кроме того, большую роль играют коррозионные процессы. Коррозия ослабляет стенки труб, делая их более уязвимыми к растрескиванию даже при незначительных механических воздействиях. Особенно опасна стресс-коррозия - разрушение материала под воздействием одновременно коррозионной среды и внутреннего напряжения.

Также к образованию трещин могут привести:

1. Механические повреждения при монтаже, транспортировке или эксплуатации;
2. Ошибки при сварке (неполный провар, пористость, шлаковые включения);
3. Технологические дефекты материала, например, наличие микропустот или включений;
4. Внешние воздействия, такие как сейсмическая активность, вибрация, осадки грунта и промерзание.

Классифицируют трещины по разным признакам:

1. По направлению: продольные, поперечные, кольцевые;

2. По глубине: поверхностные, сквозные, подповерхностные;
 3. По причине возникновения: термические, коррозионные, усталостные, технологические;
 По степени опасности: активные (растущие) и пассивные (стабилизированные)

Особую опасность представляют микротрещины, которые невозможно выявить при обычном визуальном осмотре. Они могут долгое время оставаться незамеченными, развиваясь внутри металла и распространяясь под действием нагрузок. Без своевременной диагностики такие дефекты способны привести к внезапному разрушению трубопровода.

Современные условия эксплуатации требуют регулярного и надежного контроля за состоянием трубопроводов. Однако, традиционные методы, такие как визуальный осмотр или гидравлические испытания, не всегда эффективны для обнаружения внутренних или подповерхностных трещин. Именно поэтому всё более актуальными становятся методы, основанные на анализе изображений, полученных при помощи камер, установленных внутри труб, или других средств неразрушающего контроля.

Применение таких методов позволяет обнаруживать трещины на ранней стадии, оценивать их форму, длину и направление распространения.

Это, в свою очередь, позволяет принять меры до того, как произойдет авария, тем самым повышая надёжность и безопасность всей системы.

Таким образом, изучение причин образования трещин и разработка эффективных методов их обнаружения являются важной задачей в области технической диагностики трубопроводов. Это особенно актуально в современных условиях, когда возраст многих инженерных сетей превышает нормативный срок службы, а аварии могут привести к катастрофическим последствиям как для производства, так и для экологии.

1.2 Опасность трещин в трубопроводах и необходимость раннего выявления

Возникновение трещин в трубопроводах - это не только структурное нарушение, но и потенциальный источник серьёзной техногенной и экологической угрозы. Даже небольшие по размеру дефекты могут привести к нарушению целостности системы, особенно если речь идёт о трубопроводах, по которым транспортируются опасные вещества: газ, нефть, кислоты или другие химически активные среды.

Одним из главных последствий трещинообразования является утечка транспортируемых веществ. Такая утечка может не сразу быть замечена, особенно если дефект развивается изнутри и не даёт явных внешних признаков. В промышленных условиях это грозит не только потерей продукции, но и повышением давления в системе, что усиливает разрушение и может привести к взрывам или возгораниям - особенно в случае с углеводородами [3]. В бытовых или коммунальных сетях это может вызвать аварийные отключения, подтопления, перебои с подачей воды или тепла.

Дополнительно, трещины зачастую становятся пусковым механизмом коррозии. Через микротрещины внутрь металла проникает влага, кислород и агрессивные соединения, ускоряя электрохимические процессы разрушения. В результате ухудшаются механические свойства материала, и трещина начинает расти с большей скоростью. Особенно быстро протекает этот процесс в трубах, подверженных переменным температурным и гидравлическим режимам.

Экологические последствия трещин могут быть катастрофическими. Аварии на трубопроводах с утечкой нефти или газа вызывают загрязнение почвы, отравление водных источников, гибель флоры и фауны. В городских условиях такие повреждения могут также повлечь за собой эвакуацию населения и нарушение функционирования социальной инфраструктуры.

С экономической точки зрения, последствия выражаются в:

1. потерях транспортируемого вещества;
2. затратах на устранение аварий;
3. простоев оборудования;
4. штрафах за экологические нарушения;
5. нарушении контрактных обязательств и логистических цепочек.

Серьёзность последствий особенно возрастает при позднем выявлении трещин, что делает своевременную диагностику жизненно необходимой. Однако практика показывает, что большая часть дефектов обнаруживается уже после проявления явных признаков разрушения. Это связано с ограниченностью традиционных методов контроля, таких как внешний осмотр или гидравлические испытания. Они не всегда позволяют обнаружить скрытые или подповерхностные трещины, которые, согласно , являются наиболее коварными и опасными.

Современные условия эксплуатации инженерных систем требуют перехода к прогностическому техническому обслуживанию - когда дефекты выявляются не по факту аварии, а в процессе регулярного мониторинга. Именно поэтому всё большую актуальность приобретают методы визуального контроля с элементами цифровой обработки изображений. Эти технологии позволяют анализировать состояние внутренних поверхностей труб с высокой точностью, выявляя трещины по форме, длине.

Кроме того, автоматизация анализа изображений существенно снижает влияние человеческого фактора. Алгоритмы компьютерного зрения позволяют обрабатывать данные в режиме реального времени, классифицируя трещины по степени опасности и давая возможность оперативно принимать решения о ремонте или замене участка трубопровода.

1.3 Обзор существующих методов диагностики и визуального контроля

Один из самых простых и традиционных методов - это визуальный контроль, выполняемый оператором вручную при помощи фонарей, зеркал, эндоскопов и камер. Он позволяет выявлять крупные дефекты и механические повреждения на доступных участках труб, однако сильно ограничен в эффективности при работе с внутренними поверхностями или при наличии отложений. Главный недостаток визуального осмотра - высокая зависимость от человеческого фактора и невозможность оценки скрытых повреждений.

Рисунок 1. Визуальный осмотр

Более современным способом диагностики считается видеосъемка с использованием специальных камер. Их запускают внутрь труб с помощью гибких кабелей, тросов или роботов, чтобы получить прямую трансляцию состояния внутренней поверхности. Такие системы обычно оснащены подсветкой, стабилизацией изображения, автофокусом и возможностью записи - это позволяет фиксировать трещины, коррозию, вмятины и другие дефекты, даже в труднодоступных местах. Однако, несмотря на техническую оснащенность, оценка полученных материалов по-прежнему зависит от внимательности и опыта оператора, а значит - человеческий фактор остаётся важным ограничением такого подхода. Для более точного анализа состояния материала и выявления внутренних или скрытых трещин применяются методы неразрушающего контроля (НК). Среди них:

Ультразвуковой контроль (УЗК) - используется для определения толщины стенок и обнаружения подповерхностных дефектов. Ультразвуковая

волна отражается от внутренних неоднородностей, позволяя выявить трещины, даже если они не выходят на поверхность. Ультразвуковой метод основан на анализе процесса распространения упругих колебаний в диагностируемом объекте. Ультразвуковые колебания способны отражаться от внутренних неоднородностей среды, что и является основой для данного метода [3]

Магнитопорошковый метод - эффективен при контроле ферромагнитных материалов. Основан на выявлении магнитных потоков, искаженных трещинами и другими дефектами.

Рисунок 2. Неразрушающий контроль трубопровода магнитопорошковым методом

Рисунок 3. Неразрушающий контроль трубопровода электромагнитным методом

Рисунок 4. Неразрушающий контроль трубопровода методом рентгенографии

Несмотря на эффективность перечисленных способов, они в большинстве своём остаются точечными, трудоёмкими и зависящими от условий доступа и опыта оператора. В условиях растущих требований к безопасности и масштабам трубопроводных систем всё большую актуальность приобретает автоматизированные методы визуального контроля, основанные на компьютерном зрении и алгоритмах машинного обучения. Эти технологии позволяют проводить диагностику быстро, без физического вмешательства и с минимизацией влияния человеческого фактора, что делает их перспективным направлением развития отрасли.

Глава 2. МЕТОДИКА И ИНСТРУМЕНТЫ ИССЛЕДОВАНИЯ 2.1 Выбор программных средств и инструментов анализа. Современные проблемы технического зрения, включая автоматическое нахождение трещин на поверхности трубопроводов, решаются с применением методов машинного обучения - способа, при котором алгоритмы самостоятельно обнаруживают закономерности в сведениях и принимают решения на их базе. Особую эффективность в подобных задачах продемонстрировали методы машинного и глубокого обучения, базирующиеся на применении искусственных нейронных сетей, способных анализировать изображения, извлекать признаки и выделять дефектные области с высокой точностью.

Для реализации этих методов необходимо применение специализированных программных инструментов, которые обеспечивают:

1. Обработку и подготовку изображений;
2. Построение, обучение и валидацию моделей;
3. Визуализацию результатов и оценку метрик;
4. Работу с большими объёмами данных в локальных или облачных средах.

В данной работе применялись программные средства на базе языка Python, предназначенные для реализации и обучения нейросетевых архитектур на двух различных типах датасетов: с полигональной разметкой (YOLO-формат) и с пиксельными масками (PNG-маски).

Использование двух моделей - Ultralytics YOLOv8 и DeepLabV3 - позволило провести комплексное сравнение методов обнаружения трещин по их локализации и точности сегментации. Проект реализован в локальной среде на Python 3.13 с поддержкой GPU (CUDA), что обеспечило ускоренное обучение и возможность обрабатывать большой объём изображений без потерь в производительности.

Применённые инструменты и библиотеки:

1. Ultralytics YOLOv8 - фреймворк для быстрой реализации задач сегментации на базе YOLO. Применён к датасету с координатной разметкой трещин в формате YOLO Segmentation, с обучением по маскам-контурным полигонам.
2. DeepLabV3 (via torchvision.models) - архитектура глубокого обучения для пиксельной семантической сегментации, использовалась с датасетом, где аннотации представлены в виде PNG-масок. Применяется для точного распознавания формы трещин.
3. Matplotlib и Seaborn - использовались для визуального анализа: построения графиков обучения, визуализации потерь, отображения наложенных масок и сравнения метрик.

Обучение моделей проводилось в 100 эпох с использованием оптимизатора Adam, снижением потерь (loss) и визуализацией результатов Adam (Adaptive Moment Estimation) - это распространённый алгоритм оптимизации, который активно используется в глубоком обучении и машинном обучении (ML) [5]. Он предназначен для эффективного обновления весов сети в процессе обучения путем адаптации скорости обучения для каждого параметра в отдельности. В YOLOv8 применялись также встроенные аугментации (HSV, flip, mosaic), а DeepLab - кастомные трансформации через torchvision.transforms

В качестве моделей для автоматического обнаружения трещин в изображениях трубопроводов в данной работе выбраны две нейросетевые архитектуры: YOLOv8 [7] и DeepLabV3 [8]. Эти модели представляют два разных подхода в компьютерном зрении - детекцию объектов и семантическую сегментацию - и позволяют провести разносторонний анализ применимости глубокого обучения к задаче технического контроля.

YOLOv8 - модель для детекции объектов. YOLO расшифровывается как You Only Look Once. Это широко известная архитектура компьютерного зрения, которая знаменита в том числе своим огромным количеством версий, представляет из себя целую фундаментальную модель, которую можно использовать для классификации, трекинга объектов на видео, задач pose estimation и т.д. Все это в реальном времени. [6] Главная идея YOLO заключается в том, чтобы разбить изображение на сетку и сразу предсказать координаты ограничивающих рамок (bounding boxes) и классы объектов, содержащихся в этих участках.

Рисунок 5. Архитектура внутреннего функционала YOLOv8

Основные особенности YOLOv8:

1. Одношаговая архитектура: модель одновременно определяет, что присутствует на изображении и где именно.
2. Высокая скорость инференса: особенно при использовании GPU, что делает YOLO подходящей для задач в реальном времени.
3. Улучшенные блоки C2f и backbone.
4. Универсальность: поддержка задач классификации, сегментации и детектирования в рамках единого фреймворка.

DeepLabV3 - модель для семантической сегментации

DeepLabV3 - это одна из ведущих архитектур для семантической сегментации изображений, широко применяемая в задачах, требующих точной локализации объектов на уровне пикселей. В отличие от методов, использующих ограничивающие рамки, данная модель классифицирует каждый пиксель изображения, что особенно важно при анализе дефектов сложной формы, таких как трещины.

Рисунок 6. Архитектура внутреннего функционала DeepLabV3

Технические особенности DeepLabV3:

1. Расширенные свёртки (dilated convolutions): позволяют расширить область восприятия нейрона без потери разрешения;
2. Модуль ASPP (Atrous Spatial Pyramid Pooling): извлекает признаки на нескольких масштабах одновременно, что особенно полезно при трещинах разной протяжённости и ориентации;
3. Гибкая интеграция с различными скелетами: такими как ResNet или MobileNet, что обеспечивает баланс между точностью и скоростью;
4. Пиксельная точность: каждая точка изображения получает метку, что делает модель особенно ценной в задачах промышленной дефектоскопии и контроля качества.

2.2 Подготовка и описание датасетов

Исходные данные были использованы (общедоступные изображения с трещинами на металлических и бетонных поверхностях труб, а также синтезированные изображения (для имитации условий визуального осмотра внутренней поверхности труб)). Главными критериями выбора стали:

1. Наличие ясной визуализации трещин;
2. Многообразие по размеру, форме и ориентации дефектов;
3. Наличие аннотаций или возможность их создания самостоятельно;
4. Достаточный объем выборки для обучения и проверки.

Структура первого датасета (YOLOv8)

Этот датасет был адаптирован под формат, совместимый с моделью YOLOv8. Он содержит:

Рисунок 7. Структура набора данных

Рисунок 8. Содержимое файла data.yaml для настройки набора данных

Структура набора данных включает в себя:

1. Директория `test/train/valid/images` с исходными изображениями;
2. Директория `test/train/valid/labels` с аннотациями в формате YOLO (txt-файлы). Каждая строка аннотации отображает очертания объекта в формате:

```
class_id x1 y1 x2 y2 x3 y3 ... xn yn
```

3. Файл конфигурации `data.yaml` служит для хранения данных о структуре набора данных. В данной работе он применяется для обозначения адресов к изображениям для обучения и валидации. Кроме того, в нём определены общие число классов и их названия.

Особенности:

1. Координаты нормированы по ширине и высоте изображения;
2. Вместо прямоугольника используется контур объекта - масочная сегментация, отражающая форму трещины.

Структура второго датасета (DeepLabV3) содержит:

Рисунок 9. Директория набора данных DeepLabV3

Для обучения модели DeepLabV3 применялся датасет с пиксельной сегментацией:

1. Директория `staclozeta , cracktile , membrane/images` содержит исходные изображения;
2. Директория `staclozeta , cracktile , membrane/label` - бинарные маски, где трещины - обозначены белым цветом (255), а фон - чёрным (0).

Особенности:

1. чёткая граница дефекта на уровне пикселя;
2. используются кросс-энтропийная либо Dice loss-функции при обучении.

В отличие от YOLO, DeepLabV3 управляется исключительно из Python-кода, что избавляет от необходимости в отдельном файле конфигурации. Вся настройка и параметры задаются непосредственно в коде, минуя этапы создания и редактирования внешних конфигурационных файлов.

Глава 3. ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ И СРАВНИТЕЛЬНЫЙ АНАЛИЗ

3.1 Обучение моделей

Для старта обучения модели сегментации была применена архитектура YOLOv8-seg, приспособленная под задачу пиксельной сегментации. Обучение происходит с применением написанного мной Python-скрипта, где реализована настройка всех параметров посредством объектной модели Ultralytics.

Рисунок 10. Python-скрипт для обучения модели YOLOv8

Ultralytics - это платформа с открытым исходным кодом, написанная на Python. Её основная задача - обучение и применение моделей YOLO (You Only Look Once), включая модели, способные работать с сегментацией изображений. В рамках данной работы был осуществлён запуск обучения модели YOLOv8-seg. Также была произведена настройка параметров обучения, загрузка набора данных и, как заключительный этап, оценка полученных результатов. Ultralytics предоставляет удобный API высокого уровня, что значительно облегчает интеграцию и управление всем процессом машинного обучения. Внутри функции `train()` происходит первоначальная настройка модели. Затем следует запуск процесса обучения, использующего заранее заданные гиперпараметры.

YOLOv8 автоматически генерирует графики, показывающие распределение объектов, их размеры и положения на изображениях. Эти визуализации помогают быстро оценить качество разметки и выявить ошибки ещё до начала обучения. В DeepLabV3 такой анализ по умолчанию не предусмотрен, что делает YOLO более удобным для предварительной проверки датасета.

Результаты обучения:

Рисунок 11. Распределение количества трещин в модели YOLOv8

Левый график отображает число экземпляров каждого класса. В данном случае в наборе данных присутствует всего один класс, обозначенный как `crack`. Этот класс представлен достаточно равномерно, насчитывая примерно 2500 аннотаций, что свидетельствует об одноклассовой структуре датасета.

На правом графике визуализированы все ограничивающие рамки, наложенные на условное изображение. Наблюдается явное преобладание рамок в центральной области, что указывает на тенденцию возникновения трещин преимущественно в этой зоне изображений. Дополнительно отмечается наличие повторяющихся элементов, которое, вероятно всего, обусловлено идентичной структурой изображений или сложной локализацией трещин на сгенерированных кадрах.

Рисунок 12. Изменение функций качества и метрик потерь при обучении YOLOv8

Проведя сотню итераций обучение модели YOLOv8, я отслеживал, как меняются ошибки (потери) и точность. На этих графиках показано, как с каждой эпохой ошибки уменьшаются, а точность модели становится лучше.

Результаты тренировки нейронной сети YOLOv8 демонстрируют позитивную и устойчивую тенденцию. На рисунке 12 видно, что все основные параметры потерь, отвечающие за локализацию, сегментацию и классификацию, непрерывно уменьшаются. Параллельно, показатели качества - точность, полнота, средняя точность (mAP) - постоянно увеличиваются, подтверждая эффективность процесса обучения. Отсутствие заметной разницы между результатами свидетельствует об отсутствии переобучения.

Обучение модели на втором датасете DeepLabV3:

Рисунок 13. Python-скрипт для обучения модели DeepLabV3

В ходе работы были задействованы следующие важные библиотеки:

`torch`, `torchvision` - основные инструменты для создания, обучения и обработки моделей, а также для работы с данными;

`PIL` и `transforms` - инструменты для загрузки изображений и первичной предобработки (например, изменение масштаба и преобразование в тензоры);

`tqdm` - для визуализации прогресса обучения;

`cv2` - для автоматической загрузки изображений и масок, находящихся в подпапках набора данных.

Предварительно данные были разделены на три категории, на основе которых автоматически формировались списки с путями к изображениям и маскам. Обработка изображений и соответствующих масок осуществлялась через пользовательский класс `Dataset`, подававший данные в модель посредством `DataLoader`.

Для обучения применялась модель `deeplabv3_resnet50`, адаптированная к задаче бинарной сегментации путём изменения последнего классификационного слоя.

Функция потерь была выбрана `CrossEntropyLoss`, поскольку задача представляла собой бинарную классификацию пикселей: трещина или фон.

Качество сегментации оценивалось, прежде всего, визуальным соответствием предсказанных масок исходным фрагментам.

Визуальные результаты работы моделей:

Рисунок 14. Визуально предсказанные маски модели YOLOv8

Рисунок 15. Визуально предсказанные маски модели DeepLabv3

Визуально я могу сделать вывод, что обученные модели явно имеют недостатки, но все-таки выполняют свою поставленную задачу. Поэтому далее и будет мной проведен расчет метрик для расчета точности моделей.

3.2 Сравнение YOLOv8 и DeepLabV3

Далее было проведено сравнение моделей. Цель заключалась в том, чтобы оценить, насколько точно каждая из моделей определяет расположение дефектов на изображениях. Сравнение осуществлялось на основе матриц несоответствий (`confusion matrix`), а также с расчётом ключевых метрик качества `precision`, `recall`, `IoU`, что позволило оценить эффективность алгоритмов.

Рисунок 3.2.1 Матрица несоответствий модели YOLOv8

804 изображений с трещинами модель правильно распознала (True Positive)

632 изображений без трещин модель ошибочно пометила как трещины (False Positive)

23 изображения с трещинами были пропущены - модель не нашла трещину (False Negative)

TN - True Negative (изображения без трещин, распознаны правильно) не указаны, скорее всего в тестовом наборе все изображения содержали трещины, как и у DeepLab

Расчет метрик. Для того что рассчитать следующие метрики, такие как `precision`, `recall`, `IoU`, F-1 score, я воспользовался уравнением Карла Персона[11]

$Precision(точность) = \frac{TP}{TP + FP} = 0.5598 = 56\%$

Где:

TP (True Positive) - изображения с правильно распознанными трещинами модели YOLOv8

FP (False Positive) - изображения, на которых модель YOLOv8 при классификации трещин приняла фон за трещины

Recall(полнота) = $\approx 0.9722=97\%$

TP(True Positive) - изображения с правильно распознанными трещинами модели YOLOv8
FN(False Negative)-изображения на которых модель YOLOv8 не обнаружила трещины

IoU = ≈ 0.551

P(True Positive) - изображения с правильно распознанными трещинами модели YOLOv8
FP(False Positive) - изображения, на которых модель YOLOv8 при классификации трещин приняла фон за трещины
FN(False Negative)-изображения на которых модель YOLOv8 не обнаружила трещины

F-1-score = $\approx 0.7101=71\%$

Высокий показатель recall (97%) свидетельствует о том, что YOLOv8 эффективно обнаруживает трещины - она почти никогда их не пропускает.

Низкий уровень precision (55%) говорит о тенденции модели к избыточной сегментации: она нередко выделяет не только трещины, но и элементы фона.

F1-score (71%) и IoU (55%) демонстрируют, что, хотя модель уверенно идентифицирует дефекты, необходимо улучшить точность, снизив количество ложных срабатываний.

Рисунок 3.2.2 Матрица несоответствий модели DeepLabV3

TP=1595505; FP=2312210; FN=643122; TN=105537355

Расчет метрик:

Precision(точность) = $\approx 0.408=41\%$

Recall(полнота) = $\approx 0.7129=71\%$

IoU = ≈ 0.337

F-1-score = $\approx 0.4614=46\%$

Высокое значение recall, достигающее 71%, свидетельствует о том, что алгоритм DeepLabV3, как правило, обнаруживает трещины. Однако иногда он всё же не распознаёт их.

Средний показатель precision, составляющий 41%, указывает на склонность модели к ошибкам: она нередко причисляет к трещинам элементы фона или прочие незначимые области.

Показатели F1-меры (46%) и IoU (33.7%) демонстрируют, что, хотя модель и способна находить дефекты, её точность overall и соответствие эталону оставляют желать лучшего. Эти данные подразумевают необходимость совершенствования архитектуры, оптимизации пороговых значений либо улучшения баланса классов в процессе обучения.

По итогам сравнения модель YOLOv8 продемонстрировала превосходные результаты по всем основным критериям оценки. Это делает её наиболее подходящим вариантом для автоматизированного обнаружения и сегментации трещин в ситуациях, где критически важно минимизировать пропуск дефектов. В свою очередь, DeepLabV3, хоть и отличается более сложной архитектурой, нуждается в дополнительной настройке. Это касается как обработки фоновых данных, так и оптимизации процесса обучения.

Для обеих моделей - YOLOv8 и DeepLabV3 - я бы рекомендовал использовать аугментацию данных. Аугментация, или генерирование новых данных на основе имеющихся, позволяет довольно просто и дешево решать часть проблем с обучающей выборкой подручными способами. В случае нейронных сетей, распространенным явлением стало астраивать аугментацию непосредственно в процесс обучения, модифицируя данные каждую эпоху]

Для более наглядного сопоставления я составил сравнительную таблицу, в которой указаны различия между моделями, которые я тестировал и сравнивал YOLOv8 и DeepLabV3. В таблице я показал особенности аннотаций, форматов данных, способов обработки трещин, уровня детализации, а также плюсы и минусы каждой модели. Некоторые пункты основаны на моих личных наблюдениях, полученных в процессе практического тестирования.

Таблица 1. Сравнительные характеристики моделей

Параметр	Датасет для YOLOv8	Датасет для DeepLabV3 (ResNet-50)
Тип задачи	Локализация объектов (обнаружение трещин)	Пиксельная семантическая сегментация (области трещин)
Тип аннотаций	Координаты контура трещины (полигон)	Бинарные маски (0 - фон, 255 - трещина)
Формат аннотаций	txt: список координат, образующих полигон	.png или .jpg маски, соответствующие каждому изображению
Нормализация координат	Нормированные (0-1)	Абсолютные пиксели
Обработка формы трещин	Точное приближение по координатам (полигон)	Точное описание по пикселям
Уровень детализации аннотаций	Высокий	Высокий
Размер изображений	Разный, масштабируются	Одинаковый (например, 640x640)
Формат изображений	.jpg, .png, .bmp	.png (для сохранения масок)
Совместимость с фреймворками	UltraMetrics YOLO	PyTorch, TensorFlow, DeepLabV3
Целевой формат данных	Разметка контуров объектов	Разметка по-пиксельно
Объём аннотированных данных	Высокий (координаты полигона)	Высокий (маски)
Простота подготовки	Средняя	Сложная
Обучаемая модель	YOLOv8	DeepLabV3 (ResNet-50)
Преимущества	Быстрое обучение, высокая скорость инференса	Высокая точность границ, пригодность для технического анализа

Недостатки: Сложность аннотаций, чувствительность к шуму, высокая трудоемкость разметки, медленный инференс
Преимущества: Пере или недоразделение, Пересегментация, Недоразделение
Общие выводы: Модель YOLOv8 обеспечивает быстрый анализ и точное приближение формы трещины в использовании. Модель DeepLabV3 сложна, нуждается в большем обучении.

Выводы главы:

В процессе обучения и анализа моделей сегментации YOLOv8 и DeepLabV3 для задачи детекции трещин были получены следующие результаты:

Проведено обучение модели YOLOv8 с использованием аннотаций и настроек, реализованных через Ultralytics.

Выполнено обучение модели DeepLabV3 на основе бинарных масок с применением библиотеки PyTorch.

Для оценки качества сегментации использовались метрики Карла Персона, IoU, Recall и F1-score.

Проведено сравнительное тестирование моделей по результатам матриц ошибок и метрик качества. На моем примере выявлено, что YOLOv8 показывает более высокую полноту и пригодна для задач, где важно минимизировать пропуски дефектов, тогда как DeepLabV3 требует дополнительной оптимизации для повышения точности.

ГЛОССАРИЙ

НК-неразрушающий контроль

Инференс - это процесс применения уже обученной модели к новым, ранее не виденным данным для получения предсказаний.

YOLO (You Only Look Once) - это алгоритм и библиотека для обнаружения объектов на изображениях и видео в реальном времени.

DeepLab - это семейство алгоритмов для семантической сегментации изображений

Итерация - это однократное выполнение полного цикла вычислений или действий в процессе обучения модели или при работе алгоритма.

Precision - показывает, насколько правильны предсказания модели среди тех случаев, где она "нашла трещины".

Recall - показывает, насколько модель вообще нашла все трещины, которые были на изображении.

IoU (Intersection over Union, пересечение над объединением) - показывает, насколько предсказанная область трещины совпадает с эталонной.

F1-score - это сбалансированная метрика, которая учитывает и Precision, и Recall.

Простыми словами: одна цифра, которая показывает общее качество - насколько модель хорошо и правильно находит трещины и не пропускает их.

ЗАКЛЮЧЕНИЕ

В рамках выполнения дипломной работы проведена исследовательская и практическая деятельность, целью которой стало изучение и сравнительный анализ методов визуализации и обработки изображений для обнаружения трещин в трубопроводах. Основное внимание было уделено применению современных нейросетевых архитектур, таких как YOLOv8 и DeepLabV3, для решения задач автоматического выявления и сегментации дефектов на внутренних поверхностях труб.

На теоретическом этапе были исследованы причины возникновения трещин, их классификация и потенциальные последствия в случае несвоевременного обнаружения. Также был выполнен обзор существующих методов контроля, включая визуальные, неразрушающие и автоматизированные подходы, основанные на технологиях компьютерного зрения.

Практическая часть включала в себя отбор и подготовку нескольких наборов данных с изображениями трещин, а также обучение моделей YOLOv8 и DeepLabV3. Каждая модель была обучена на специально подготовленных данных, предварительно размеченных и прошедших обработку.

В результате проведенного сравнения установлено, что обе модели успешно решают задачу обнаружения трещин, однако демонстрируют различную специфику: YOLOv8 обеспечивает высокую скорость инференса и оптимальна для задач в реальном времени, в то время как DeepLabV3 демонстрирует более точное определение геометрии дефектов благодаря пиксельной сегментации. Также отмечено преимущество YOLOv8 в плане удобства подготовки и анализа аннотаций.

Анализ показал, что применение нейросетевых методов значительно повышает эффективность раннего выявления дефектов в трубопроводах.

В качестве перспектив дальнейшего развития предлагается увеличить объем обучающей выборки, использовать более специализированные архитектуры сегментации, а также интегрировать обученные модели в реальные системы инспекции с использованием роботизированных платформ и камер высокого разрешения.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

[1] Трубопровод: типы, применение и особенности <https://re-st.ru/interesnoe/truboprovod-typy-primeneniye-i-osobennosti/>?utm_source=chatgpt.com

[2] **PIPE CRACKS DETECTION METHODS - A REVIEW**
https://ijomae.com/wp-content/uploads/2023/01/pag-114-119_PIPE-CRACKS-DETECTION-METHODS.pdf

[3] ISSN 2072-0297 Мониторинговый журнал 185-187 <https://moluch.ru/archive/102/23388/>

[4] Что такое машинное обучение? <https://aws.amazon.com/ru/what-is/machine-learning/#text=D0%9C%D0%B0%D1%88%D0%B8%D0%BD%D0%BD%D0%BE%D0%B5%20%D0%BE%D0%B1%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5%20%E2%80%93%20%D1%8D%D1%82%D0%BE%20%D0%BD%D0%B0%D1%83%D0%BA%D0%B0%20%D0%BE%D0%BD%D0%B0%20%D1%88%D0%B0%D0%B1%D0%BB%D0%BE%D0%BD%D1%8B%20%D0%B8%20%D0%BB%D0%BE%D0%B3%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B5%20%D0%B2%D1%8B%D0%B2%D0%BE%D0%B4%D1%8B>

[5] Адам Оптимизатор <https://www.ultralytics.com/glossary/adam-optimizer>

[6] Истории YOLO <https://habr.com/ru/articles/865834/>

[7] Pipeline Crack Detection Computer Vision Project <https://universe.roboflow.com/masimba/pipeline-crack-detection-8/surface-crack-detectionhttps://github.com/arthurflor23/surface-crack-detection>

- [9] YOLOv8 to YOLO11: A Comprehensive Architecture In-depth Comparative Review <https://arxiv.org/pdf/2501.13400>
- [10] A Guide to Using DeepLabV3 for Semantic Segmentation <https://www.datature.io/blog/a-guide-to-using-deeplabv3-for-semantic-segmentation>
- [11] Understanding Confusion Matrix <https://communities.sas.com/15/SAS-Communities-Library/Understanding-Confusion-Matrix/a-p/614884>