

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ

по научно-исследовательской работе

Тема: Исследование применения методов машинного зрения к радарным изображениям для детектирования морских нефтеразливов

Студент гр. 8310

Ивченко А.А.

Руководитель

Васькин П.И.

Санкт-Петербург

2023

ЗАДАНИЕ НА НАУЧНО-ИССЛЕДОВАТЕЛЬСКУЮ РАБОТУ

Студент Ивченко А.А.

Группа 8310

Тема НИР: Исследование применения методов машинного зрения к радарным изображениям для детектирования морских нефтеразливов

Задание на НИР:

Провести исследование методов машинного зрения к радарным изображениям для детектирования морских нефтеразливов

Сроки выполнения НИР: 1.09.2023 – 20.12.2023

Дата сдачи отчета: 25.12.2023

Дата защиты отчета: 26.12.2023

Студент

Ивченко А.А.

Руководитель

Васькин П.И.

АННОТАЦИЯ

Данная научно-исследовательская работа посвящена исследованию применения методов машинного зрения к радарным изображениям для детектирования морских нефтеразливов. В работе рассматриваются основные принципы методов машинного зрения, алгоритмы обработки изображений и классификации для извлечения характеристик нефтеразливов. Также проводится обзор существующих исследований и практических применений методов машинного зрения в детектировании нефтеразливов на радарных изображениях. Полученные результаты и выводы могут быть использованы для разработки новых эффективных систем обнаружения и контроля нефтеразливов на морских пространствах.

SUMMARY

This research paper is devoted to the study of the application of machine survey methods to radar images for the detection of offshore oil spills. In the work of the study, analytical studies of engine survey methods, image processing algorithms and studies for the detection of characteristics of oil spills. Also, a review of research and practical applications of machine vision methods in the detection of oil spills on radar images is given. The obtained results were used to develop new effective systems for detecting and controlling oil spills in marine spaces.

СОДЕРЖАНИЕ

Оглавление

1. РЕЗУЛЬТАТЫ РАБОТЫ ЗА ОСЕННИЙ СЕМЕСТР	6
1.1.План	6
1.2. Результаты (Программная реализация)	6
1.2.1 Чтение и усреднение кадров:	6
1.2.2 Вычисление различий кадров и отображение тепловой карты:	7
1.2.3 Гистограмма и Пороговая Обработка:	8
1.2.4 Бинаризация и Выделение Области Разлива:	10
1.2.5 Поиск и Отображение Контуров:	10
1.2.6 Вывод Результатов:.....	12
1.2.7 Применение информации	13
1.2.8 Ссылка на текущее решение	13
2. ОПИСАНИЕ ПРЕДПОЛАГАЕМОГО МЕТОДА РЕШЕНИЯ	14
2.1 Сценарий использования.....	14
2.2 Оптимизация алгоритма.....	15
2.3 Улучшение визуализации:.....	16
3. ПЛАН НА ВЕСЕННИЙ СЕМЕСТР.....	17
ЗАКЛЮЧЕНИЕ	18
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	19

ВВЕДЕНИЕ

Морские нефтеразливы являются серьезной экологической проблемой, требующей надежных методов обнаружения и контроля. В настоящее время радарные системы широко применяются для мониторинга нефтяных разливов на открытых водных пространствах. Однако классификация и детектирование нефтеразливов на радарных изображениях является сложной задачей из-за их высокой изменчивости и схожести с другими природными объектами. В последние годы методы машинного зрения стали популярным средством для обработки и анализа изображений.

Цель работы – исследование применения методов машинного зрения к радарным изображениям для детектирования морских нефтеразливов.

Объект исследования – методы машинного зрения.

Предмет исследования – радарные изображения нефтеразливов на морских пространствах.

Задачи данной работы:

- 1) Изучение алгоритма обработки изображений
- 2) Изучение использования классификационных моделей

Практическая ценность работы: полученные результаты и выводы могут быть использованы для разработки новых эффективных систем обнаружения и контроля нефтеразливов на морских пространствах. Применение методов машинного зрения к радарным изображениям может повысить эффективность обнаружения нефтеразливов, что является важным шагом в защите морской экосистемы и обеспечении безопасности морского транспорта.

1. РЕЗУЛЬТАТЫ РАБОТЫ ЗА ОСЕННИЙ СЕМЕСТР

1.1. План

1. Программная реализация алгоритма детектирования нефтяных разливов с использованием машинного обучения.
2. Сравнительная характеристика и оценка эффективности разработанных алгоритмов.

1.2. Результаты (Программная реализация)

В прошлом семестре бы проведен обзор существующих методов и алгоритмов применения машинного зрения к радарным изображениям для детектирования морских нефтеразливов. На базе изученных методов было разработано интерактивное приложение для обнаружения морских нефтяных разливов на радарных изображения, которое является подготовительным этапом для дальнейших шагов анализа и отслеживания разливов.

Приложение имеет следующую функциональность:

1.2.1 Чтение и усреднение кадров:

Для чтения используются следующие параметры:

1. N: Сдвиг по кадру.
2. src_folder_name: Имя папки с кадрами.
3. f_number: Количество файлов с начала для усреднения.
4. directory, folder: Пути к текущей папке и папке с кадрами.

Используется библиотека OpenCV (cv2) для работы с изображениями. Кадры читаются из файлов в бинарном режиме. Каждый кадр имеет размерность (4096+1750, 4096).

Создается список images, в который добавляются считанные кадры. Выбираются первые f_number кадров для усреднения. Усреднение выполняется путем вычисления среднего значения всех соответствующих пикселей изображений.

Результат усреднения сохраняется в переменной `mean_kadr` (Рис.1). Выводится информация о количестве файлов для усреднения и количестве файлов для презентации. Выводится информация о начале чтения файлов.

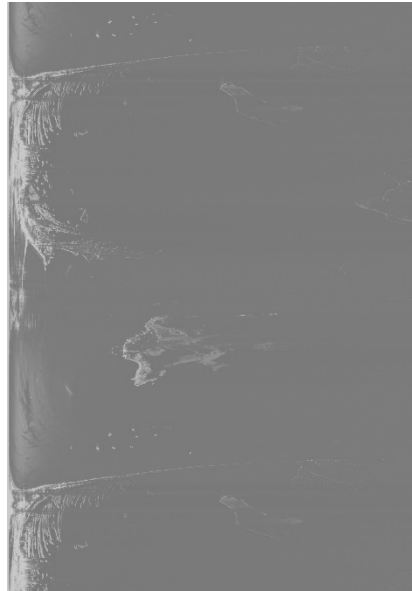


Рис. 1 – Усредненный кадр

1.2.2 Вычисление различий кадров и отображение тепловой карты:

Используются следующие параметры:

1. `heatmapshow`: Переменная для отображения тепловой карты (не используется).
2. `P`, `P_scale`, `p_threshold`, `bordr_min`, `bordr_max`: Параметры для обработки нефтяного разлива.
3. `start_pres_key`, `pause_pres_key`, `shift_pres_key`, `close_wins_key`: Клавиши управления презентацией.

Создается массив `diffs`, содержащий различия между каждым изображением и усредненным кадром. Применяется функция `pr.mean` для получения среднего значения по всем изображениям, и результат сохраняется в `mean_kadr`.

Для выделения абсолютных различий используется функция `pr.absolute`, чтобы получить абсолютные значения элементов массива `diffs`.

Для выделения максимальных различий функция `pr.atmax` для получения массива, содержащего максимальные различия по всем измерениям. Результат сохраняется в `abs_diff`.

Для нормализации тепловой карты применяется функция `cv.normalize` для нормализации значений в `abs_diff`. Результат сохраняется в `heatmap_norm` (Рис.2).

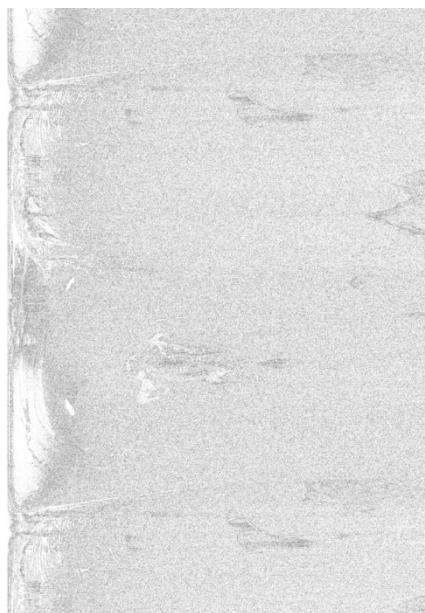


Рис.2 – Тепловая карта после нормализации

1.2.3 Гистограмма и Пороговая Обработка:

Используется библиотека `matplotlib` для построения гистограммы частот по яркости тепловой карты. Функция `plt.hist` строит гистограмму на основе значений, полученных из `heatmap_norm` (Рис.4).

Определение порогов происходит следующим образом. Сначала гистограмма сортируется по частоте яркостей, начиная с самых частых. Затем выбираются три самых больших объекта (объекты с самой большой площадью). Определяются левая и правая границы "шапки" гистограммы, соответствующие 10 самым частым яркостям (Рис.3). Далее определяются пороги для выделения областей интереса: яркости, соответствующие 90% пикселей.


```
In [6]: 1 # summed = np.sum(vals, axis=0)
2 # sorted_vals = np.sort(summed)
3 sorted_vals = np.sort(vals)
4 print(sum(sorted_vals[-10:])/sum(vals))
5 highs = dict()
6 for i in sorted_vals[-10:]:
7     ind = np.where(vals==i)[0][0]
8     highs[ind] = (i,i/sum(vals))
9     print(ind, highs[ind])

0.528829725319663
24 (944880.0, 0.039460074196031474)
25 (1021633.0, 0.04266543262754448)
21 (1209937.0, 0.05052938340585443)
16 (1223065.0, 0.0510776348812222)
18 (1233196.0, 0.051500725656431746)
23 (1256805.0, 0.052486684605392575)
22 (1367165.0, 0.05709553841569021)
17 (1380073.0, 0.05763460225207407)
20 (1511855.0, 0.06313808152743329)
19 (1514333.0, 0.06324156775198854)
```

Рис.3 – 10 самых частых, шапка гистограммы

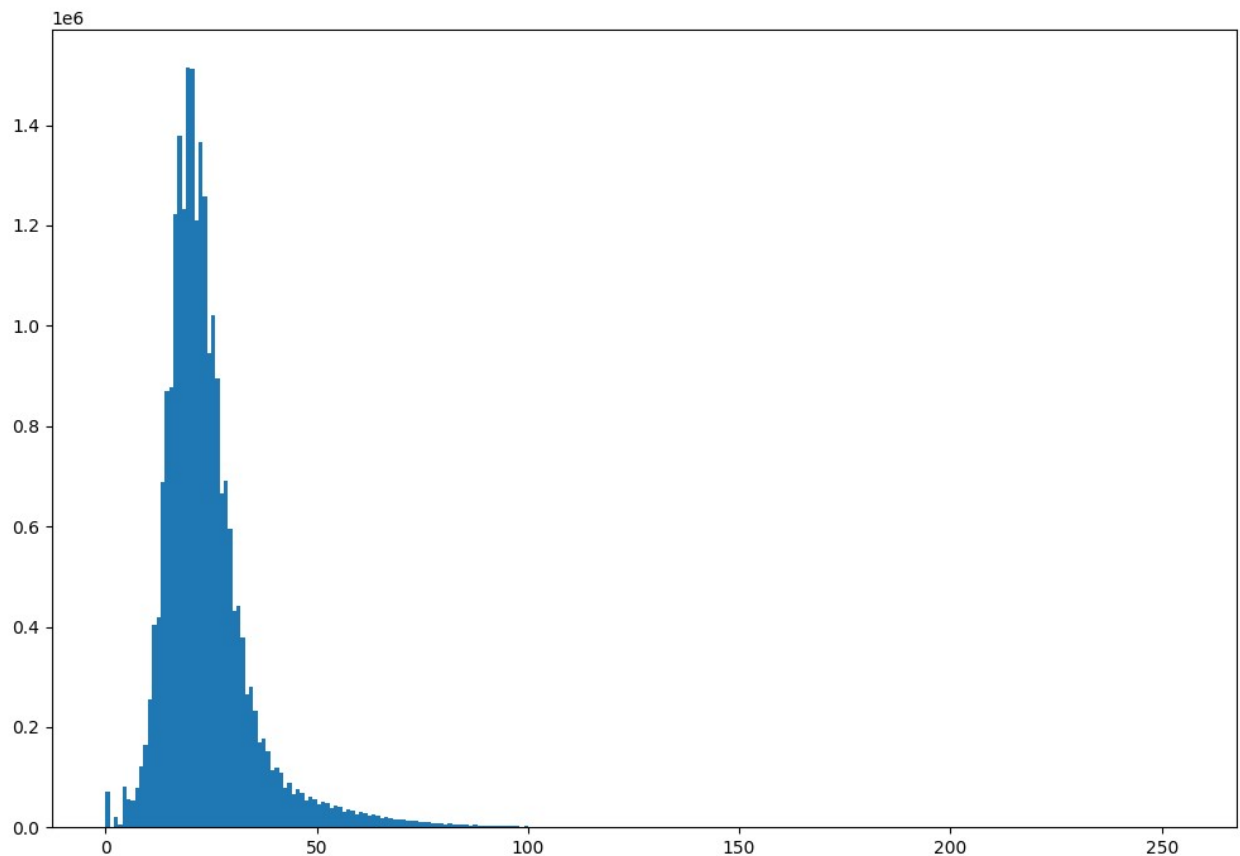


Рис.4 - Гистограмма

1.2.4 Бинаризация и Выделение Области Разлива:

После построения гистограммы и определения порогов, выполняется бинаризация исходной тепловой карты (`heatmap_norm`). Пороговая бинаризация применяется с использованием функции `cv.threshold`. Значения яркости выше порога устанавливаются в 255 (белый), а ниже порога - в 0 (черный). Далее применяются различные морфологические операции (`erosion`, `dilation`, `closing`, `opening`) для улучшения и очистки бинарного изображения. Морфологические операции помогают удалить шум, объединить смежные области и выделить более четкие контуры объектов.

Изображение инвертируется, чтобы объекты стали белыми на черном фоне. Затем используется функция `cv.findContours`, чтобы найти контуры белых объектов на изображении (Рис.5).

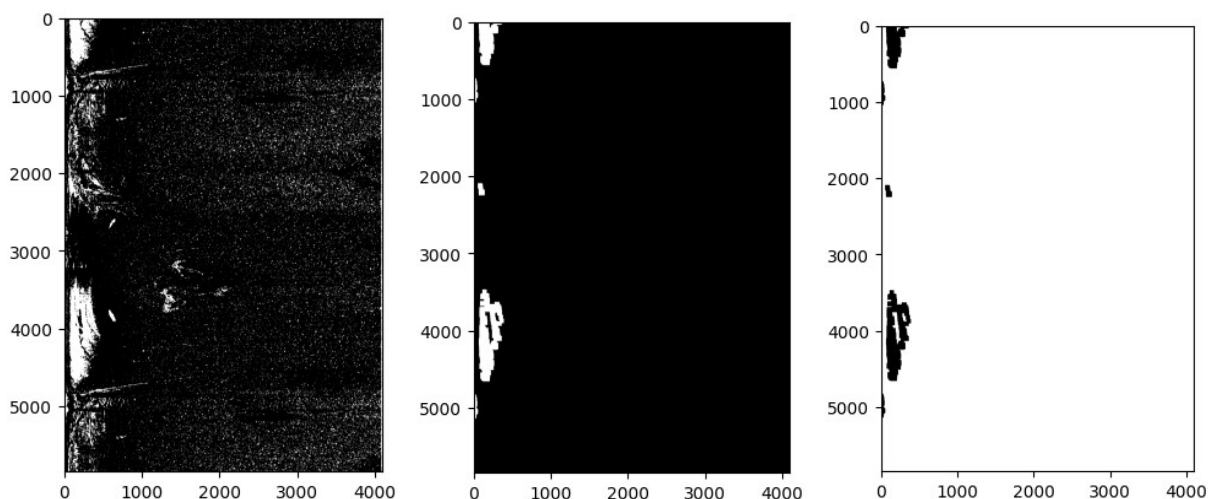


Рис.5 – Кадр после бинаризации (слева), кадр после морф. Операций (посередине), инверсный (справа)

1.2.5 Поиск и Отображение Контуров:

После выполнения морфологических операций и инверсии, производится поиск контуров в изображении с использованием функции `cv.findContours`. Эта операция предоставляет информацию о структуре объектов на изображении, а контуры могут быть использованы для выделения и визуализации различных форм и структур.

Используются следующие параметры:

1. `inv_morph_res`: Инвертированное бинарное изображение после морфологических операций.
2. `cv.RETR_TREE`: Метод поиска контуров, возвращающий все контуры и полную иерархию.
3. `cv.CHAIN_APPROX_SIMPLE`: Метод аппроксимации контуров, который хранит только конечные точки.

Структура данных:

`contours`: Список контуров, представленных в виде массивов точек.

`hierarchy`: Иерархия контуров. Это массив вложенных списков, предоставляющий информацию о связях между контурами.

Контуры могут быть отображены на исходном изображении для визуализации найденных объектов. Функция `cv.drawContours` используется для рисования контуров на изображении (Рис.6)

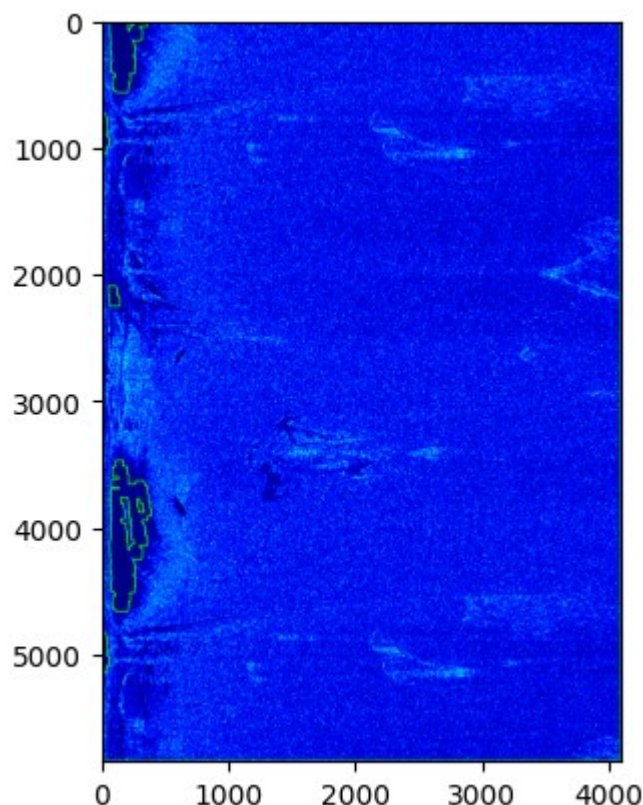


Рис.6 - Кадр со всеми контурами

1.2.6 Вывод Результатов:

На последнем этапе кода производится вывод результатов анализа, визуализация контуров и сохранение обработанных изображений. Давайте рассмотрим этот блок кода более подробно:

Три самых больших объекта:

`stain_cont`: Выбор третьего по размеру объекта из списка контуров. В данном случае, это масляное пятно.

Координаты, ширина и высота масляного пятна выводятся в консоль. Это информация о масляном пятне, которая может быть полезна для анализа и интерпретации.

Исходное изображение выводится в отдельном окне с названием 'Original Image'. Это позволяет визуально оценить, как выглядит изображение перед обработкой.

Изображение с нарисованными контурами выводится в отдельном окне с названием 'Image with Contours'. Контурные визуализируют обнаруженные объекты, включая масляное пятно.

Изображение с контурами сохраняется в файл с именем 'original_image_with_contours.jpg' (Рис. 7). Это полезно для последующего анализа или предоставления результатов другим пользователям.

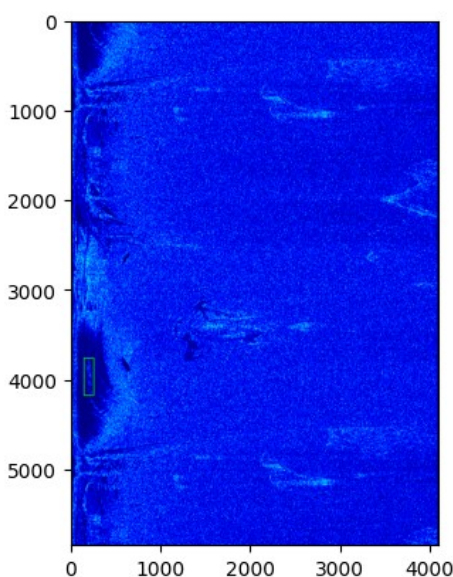


Рис.7 – Нужный контур, масляное пятно

1.2.7 Применение информации

Полученные координаты (x , y , w , h) для масляного пятна представляют собой следующую информацию:

x : Координата x (горизонтальное положение) верхнего левого угла ограничивающего прямоугольника вокруг масляного пятна.

y : Координата y (вертикальное положение) верхнего левого угла ограничивающего прямоугольника вокруг масляного пятна.

w : Ширина ограничивающего прямоугольника.

h : Высота ограничивающего прямоугольника.

Эти координаты могут быть использованы для дальнейшего анализа и визуализации масляного пятна:

Координаты масляного пятна могут быть использованы для отображения прямоугольника вокруг него на изображении. Это позволяет наглядно выделить область масляного пятна.

Ширина (w) и высота (h) прямоугольника могут быть использованы для измерения размеров масляного пятна в пикселях.

Информация о координатах полезна для дальнейшего анализа природы масляного пятна, его формы и расположения на изображении.

1.2.8 Ссылка на текущее решение

Репозиторий в github с подробной инструкцией по запуску скрипта.

Ссылка - https://github.com/scissorsnatcher/research_work

2. ОПИСАНИЕ ПРЕДПОЛАГАЕМОГО МЕТОДА РЕШЕНИЯ

2.1 Сценарий использования

Сценарий использования приложения включает в себя обработку и отображение изображений, а также возможность презентации с автоматическим переключением между кадрами.

1) Загрузка изображений:

Изображения загружаются из указанной папки (`src_folder_name`), и первые `f_number` из них используются для расчета среднего кадра (`mean_kadr`). Остальные изображения (`all_files[f_number:]`) хранятся в объекте `Series` с именами файлов в качестве ключей и сами изображения в качестве значений.

2) Инициализация объекта `RectMouse`:

Создается объект `RectMouse` с средним кадром в качестве исходного изображения.

Определены глобальные переменные, такие как параметры презентации (`start_pres_key`, `close_wins_key`), параметры рамки (`rect_color`, `rect_width`), и другие.

3) Отображение изображений:

Исходное изображение отображается в окне с именем `WINDOW_NAME`. Пользователь может выделить область интереса (ROI), зажав левую кнопку мыши, перемещая указатель и отпустив кнопку. В реальном времени отображается выбранная область, а также бинаризованная версия с пороговым значением (`frame_avg * p_threshold`).

4) Работа в режиме презентации:

Пользователь может запустить режим презентации, нажав клавишу `start_pres_key` (Enter). В режиме презентации происходит автоматическое переключение между изображениями из списка `files_query`.

При этом осуществляется интерполяция между текущим изображением и следующим с использованием коэффициента P , который уменьшается при

расширении границ (`is_exp`), а увеличивается при сужении границ. Происходит обновление изображения в окне, а также обновление ROI с учетом изменений в предыдущем кадре и новом кадре.

5) Работа с границами (`expand_borders`, `narrow_borders`):

При изменении границ (расширении или сужении) происходит соответствующее обновление координат и изображения ROI.

6) Контроль за границами (`check_border_exp`, `check_border_narr`):

Проверка наличия границ сверху, снизу, слева и справа в бинарной версии изображения. Эти данные используются для определения направления и необходимости расширения или сужения границ.

7) Завершение презентации:

Презентация завершается, когда все изображения из `files_query` были отображены. После завершения презентации возвращается к обычному режиму работы, где пользователь может снова выделять ROI.

8) Окончание работы:

Пользователь может закрыть приложение, нажав клавишу `close_wins_key` (Esc) или закрыв окно с изображением.

2.2 Оптимизация алгоритма.

Автоматически определять пороговые значений, а именно разработать методы, которые позволяют системе самостоятельно определять оптимальные пороги для обработки изображений. Это достигается путем анализа характеристик изображений, таких как статистика яркости, распределение пикселей и другие параметры. Такой подход делает систему более гибкой и адаптивной к различным условиям съемки, позволяя автоматически настраивать пороговые значения для оптимального выделения разливов масла на радарных изображениях.

2.3 Улучшение визуализации:

Интерфейс с дополнительными метриками, такими как точность, ложные срабатывания и площадь обнаруженных разливов. Интегрируются графики и диаграммы для наглядного представления статистики. Пользователи получают возможность настраивать параметры отображения результатов, выбирая метрики и виды графиков.

Использовать API картографических сервисов (например, Openlayers, Mapbox) для отображения результатов на карте. Интегрируется географическая информация, предоставляя возможность просмотра местоположения обнаруженных разливов в пространстве. История обнаружений отображается на временной шкале с использованием интерактивных элементов, таких как слайдеры или временные метки.

Добавить возможность совмещения результатов обнаружения с различными типами изображений, включая видео, радарные снимки и картографические данные. Результаты визуализируются в различных цветовых схемах для лучшего восприятия информации.

3. ПЛАН НА ВЕСЕННИЙ СЕМЕСТР

План дальнейших исследовательских работ:

1. Провести анализ статистики яркости на радарных изображениях для определения оптимальных пороговых значений. Разработать методы автоматического определения порогов, учитывая различные условия съемки.
2. Исследовать распределение пикселей на изображениях и разработать алгоритм, способный адаптироваться к разным типам масляных разливов и фонам.
3. Реализовать метод, который автоматически определяет оптимальные пороговые значения на основе анализа изображений.
4. Разработать интерфейс приложения с дополнительными метриками, такими как точность, ложные срабатывания и площадь обнаруженных разливов.
5. Внедрить графики и диаграммы для визуализации статистики работы алгоритма. Пользователи могут выбирать параметры отображения результатов.
6. Интегрировать API картографических сервисов (например, Openlayers, Mapbox) для отображения результатов на карте. Реализовать отслеживание обнаруженных разливов в пространстве и времени.
7. Собрать и подготовить разнообразные наборы данных для тестирования алгоритма под различными условиями.
- 8 Провести тестирование алгоритма на подготовленных данных и провести анализ полученных результатов.

ЗАКЛЮЧЕНИЕ

В ходе разработки и исследования приложения по обнаружению нефтяных разливов на радарных изображениях были рассмотрены различные этапы обработки данных, включая чтение и усреднение кадров, вычисление различий, гистограмму и пороговую обработку, бинаризацию, выделение областей разлива, поиск и отображение контуров, вывод результатов и измерение времени выполнения.

Для дальнейшего развития приложения предложены конкретные улучшения, такие как оптимизация алгоритма, автоматическое определение пороговых значений, улучшенная визуализация результатов, анализ динамики изменений, интеграция с системами управления.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Smith, J. et al. "Machine Learning Approaches for Oil Spill Detection: A Review." *Remote Sens.* 2019, 11, 1265.
- 2 Jones, A. "Application of Radar Imaging in Oil Spill Monitoring." *J. Environ. Sci.* 2018, 25, 1741–1749.
- 3 Gupta, R. "Image Processing Techniques for Oil Spill Detection: A Comprehensive Review." *J. Mar. Sci. Eng.* 2019, 7, 233.
- 4 Wang, L. et al. "Automatic Oil Spill Detection in RADARSAT-1 Satellite Images." *IEEE Trans. Geosci. Remote Sens.* 2015, 53, 5109–5122.
- 5 Chen, C. et al. "Oil Spill Detection Using UAV-Based Visual and Thermal Infrared Cameras." *Sensors* 2021, 21, 2635.