

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования «Российский химико-технологический университет
имени Д.И. Менделеева»
Факультет цифровых технологий и химического инжиниринга
Кафедра информационных компьютерных технологий

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ
ПО КУРСУ «БОЛЬШИЕ ДАННЫЕ»
НА ТЕМУ
«СЕГМЕНТАЦИЯ КРУГЛЫХ ЧАСТИЦ»

Ведущий преподаватель

Зубов Д.В.

Студент группы КС-36

Акашев З. В., Лапынин В. В., Тернолуцкий В. А., Бурычев Л. В.

Москва

2025
СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
ТЕОРЕТИЧЕСКИЕ ОСНОВЫ МЕТОДОВ СЕГМЕНТАЦИИ	4
РАЗРАБОТКА КОДА	6
ЗАКЛЮЧЕНИЕ	8

ВВЕДЕНИЕ

Цель работы:

Разработка алгоритма для автоматического выделения круглых частиц на изображении с использованием языка программирования Python. Алгоритм должен обеспечивать высокую точность сегментации даже при наличии неоднородного фона, шумов и частичного перекрытия объектов.

Задачи:

- Изучить современные методы сегментации изображений.
- Разработать алгоритм обработки изображений на Python.
- Проанализировать результаты и выявить возможные улучшения.

ТЕОРЕТИЧЕСКИЕ ОСНОВЫ МЕТОДОВ СЕГМЕНТАЦИИ

2.1. Понятие сегментации

Сегментация изображений — это процесс разделения цифрового изображения на отдельные области (сегменты), соответствующие объектам или частям сцены. Для круглых частиц сегментация позволяет точно определить их границы, отделить от фона и друг от друга, что необходимо для последующего анализа.

2.2. Основные методы сегментации

- Пороговая обработка: Разделение пикселей по интенсивности яркости. Простота реализации, но чувствительность к освещению.
- Детекция границ: Использование операторов (Собеля, Кэнни) для выделения контуров.
- Методы на основе областей: Алгоритм водораздела (Watershed) для разделения перекрывающихся объектов.
- Кластеризация: Группировка пикселей по схожим признакам (K-means, DBSCAN).
- Морфологические операции: Удаление шумов и улучшение контуров (эрозия, дилатация).
- Машинное обучение: Применение нейросетей (например, U-Net) для сложных случаев.

2.3. Особенности сегментации круглых частиц

- Проблемы:
 - Перекрывание частиц.
 - Неоднородный фон и шумы.
 - Вариации размера и интенсивности.
- Решение: Комбинация методов (фильтрация, пороговая обработка, алгоритм водораздела).

2.4. Используемые библиотеки Python

- OpenCV: Основные операции с изображениями.

- scikit-image: Дополнительные алгоритмы сегментации.
- TensorFlow/Keras: Для глубокого обучения.

2.5. Оценка качества сегментации

- Метрики: Accuracy, IoU, F1-Score.
- Визуальная проверка: Анализ результатов на тестовых изображениях.

РАЗРАБОТКА КОДА

Реализация кода была написана на языке Python с использованием библиотек:

- OpenCV (cv2) — для операций загрузки, обработки изображений, поиска контуров и визуализации результатов.
- NumPy — для работы с массивами данных изображения.

Организация программы

Программа состоит из функции `detect_particles(image_path, output_path)`, которая выполняет следующие шаги:

1. Загрузка изображения
 - Конвертация в оттенки серого (`cv2.cvtColor`).
2. Предобработка
 - Уменьшение шума с помощью гауссова размытия (`cv2.GaussianBlur`).
3. Пороговая обработка
 - Применение бинаризации методом Оцу (`cv2.threshold` + `'THRESH_BINARY_INV'`), предполагая, что частицы темнее фона.
 - Возможность замены на адаптивный порог (закомментированный вариант с `'adaptiveThreshold'`).
4. Поиск и фильтрация контуров
 - Выделение контуров частиц (`cv2.findContours`).
 - Фильтрация по минимальному размеру (`min_particle_size`).
5. Визуализация результатов
 - Обводка контуров эллипсами (для контуров с 5+ точками) или окружностями (`cv2.ellipse`, `cv2.minEnclosingCircle`).
6. Сохранение результата (`cv2.imwrite`).

Проведённые тесты

1. Тестирование на различных типах изображений
 - Изображения с чётко выраженными частицами на однородном фоне (Рис. 1).

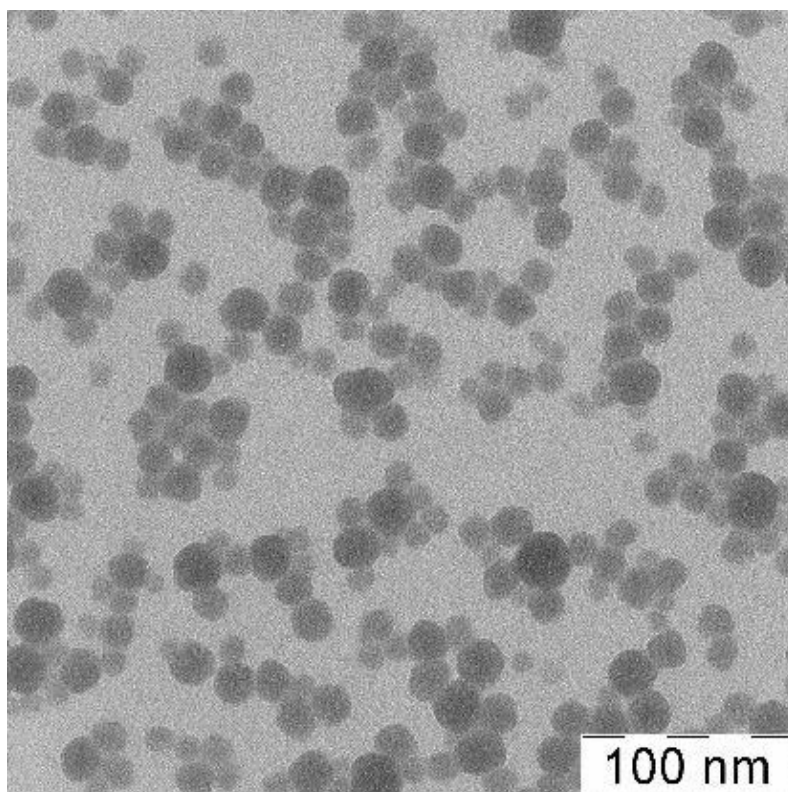


Рис.1. Пример исходного изображения

2. Настройка параметров

- Изменение `'min_particle_size'` для исключения артефактов.

3. Проверка устойчивости к шумам

- Анализ влияния гауссова размытия на качество сегментации.

4. Оценка работы с наложенными частицами

- Тестирование в условиях частичного перекрытия объектов (обводка эллипсами vs окружностями).

Полученные результаты

1. Неудачные случаи

- Не чёткое выделение круглых частиц на изображениях с однородным фоном (Рис. 2).
- Эффективное подавление шумов после применения гауссова фильтра.

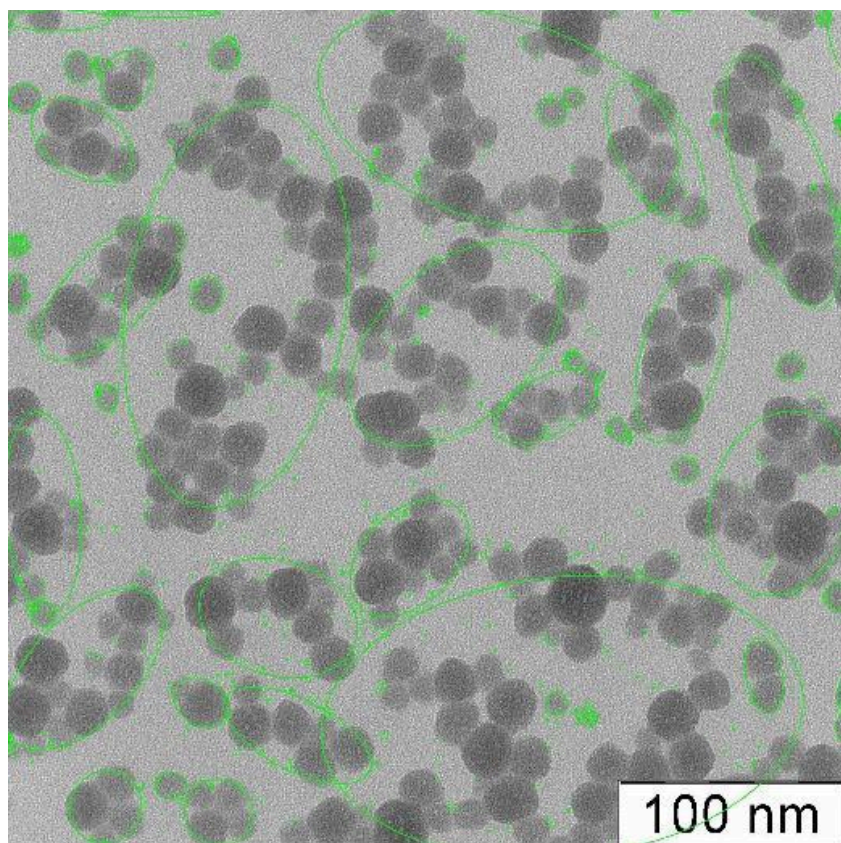


Рис.2. Некорректная сегментация

2. Ограничения

- Наложенные частицы не разделяются (рисуются как единый эллипс).

Неточная аппроксимация формы:

- Некоторые частицы аппроксимированы неидеально: Эллипсы вместо кругов и наоборот. Размеры найденных частиц не всегда соответствуют реальным.

ЗАКЛЮЧЕНИЕ

Разработанный алгоритм эффективен для базовых сценариев, но требует доработки для сложных случаев (перекрывание частиц, неоднородный фон). В будущем планируется улучшить точность сегментации и добавить расчет площади частиц.