**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**Факультет прикладной математики – процессов управления**

**отчет**

**по лабораторной работе**

**по дисциплине «Функциональное программирование»**

**на тему «Параллельная обработка»**

|  |  |  |
| --- | --- | --- |
| Студент гр. 22.Б15 |  | Агишев А.Б. |
| Преподаватель |  | Киямов Ж.У. |

**Санкт-Петербург**

**2023 г.**

**Содержание**

[1. Цель работы 2](#_Toc150296671)

[2. Задача 2](#_Toc150296672)

[3. Теоретическая часть 2](#_Toc150296673)

[4. Алгоритм метода 3](#_Toc150296674)

[5. Описание программы 4](#_Toc150296675)

[6. Рекомендации пользователю 6](#_Toc150296676)

[7. Рекомендации программисту 6](#_Toc150296677)

[8. Контрольный пример 6](#_Toc150296678)

[9. Заключение 7](#_Toc150296679)

# **Цель работы**

Разработать программу для анализа космических данных с использованием параллельных вычислений, с целью сбора статистики.

# **Задача**

1. Приобрести и закрепить знания в области компьютерного зрения и параллельных вычислений.
2. Собрать базу данных космических изображений.
3. Разработать программное обеспечение для анализа космических данных.
4. Оптимизировать вычисления и алгоритмы для ускорения обработки данных.
5. Создать интерфейс для программы.

# **Теоретическая часть**

*Компьютерное зрение (OpenCV):*

Компьютерное зрение — это область искусственного интеллекта и компьютерной науки, которая занимается обработкой и анализом изображений и видео. OpenCV (Open Source Computer Vision Library) является одной из наиболее популярных и мощных библиотек для разработки приложений в области компьютерного зрения. Важные аспекты OpenCV включают:

* Загрузка и обработка изображений и видео.
* Выделение объектов на изображениях (сегментация).
* Детекция объектов и паттернов (например, лица, автомобили).
* Измерение характеристик объектов (например, размер, яркость).

*Параллельные вычисления (threading):*

Параллельные вычисления — это методика выполнения вычислительных задач с использованием множества параллельных потоков или процессов. Это позволяет увеличить производительность и эффективность обработки данных, особенно в случае, когда задачи могут быть разделены на независимые подзадачи.

В контексте анализа космических данных и компьютерного зрения параллельные вычисления могут быть применены для следующих целей:

* Разделение обработки изображений на несколько потоков: это позволяет обрабатывать несколько изображений одновременно, ускоряя процесс анализа большого объема данных.
* Параллельные алгоритмы: разработка алгоритмов, спроектированных для эффективной параллельной обработки данных, что позволяет ускорить вычисления.

Итак, объединение компьютерного зрения с параллельными вычислениями позволяет эффективно анализировать изображения и обрабатывать большие объемы данных, что особенно важно в контексте задачи анализа космических данных.

# **Алгоритм метода**

1. Считывание всех изображений из заданной директории.
2. Параллельная обработка всех выбранных изображений, а также в случае большого изображения его параллельная обработка с предварительным преобразованием его на составные части.
3. Преобразование каждого изображения в оттенки серого.
4. Применение пороговой фильтрации для создания двоичного изображения для каждого изображения.
5. Поиск контуров на двоичном изображении для каждого изображения.
6. Классификация на звезды и планеты на основе площади контура для каждого изображения.
7. Сохранение обработанного изображения с выделенными контурами в заданную папку для каждого изображение.
8. Вывод статистики для каждого изображения.

# **Описание программы**

В программе используется 2 класса: 1 связанный с интерфейсом программы, 1 связанный с обработкой изображений. В таблице 5.1 представлено описание классов.

*Таблица 5.1. Описание классов*

|  |  |  |
| --- | --- | --- |
| Имя класса | Наследование | Описание |
| ImageAnalysis | — | Поиск объектов путем использования компьютерного зрения, сбор статистики |
| ImageAnalysisApp | — | Создание интерфейса, распараллеливание обработки изображений |

Описание переменных класса *«ImageAnalysis»* представлено в таблице 5.2.

*Таблица 5.2. Описание переменных класса «ImageAnalysis»*

|  |  |  |
| --- | --- | --- |
| Тип | Имя | Описание |
| int | stars\_count | Количество звёзд на изображении |
| int | planets\_count | Количество планет на изображении |
| str | image\_path | Путь до изображения на устройстве |
| numpy.ndarray | image | Загруженное изображение в виде массива NumPy |
| numpy.ndarray[] | images | Составные части большого изображения |

Описание функций класса *«ImageAnalysis»* представлено в таблице 5.3.

*Таблица 5.3. Описание функций класса «ImageAnalysis»*

|  |  |  |
| --- | --- | --- |
| Входные данные | Имя | Описание |
| Путь до изображения на устройстве | \_\_init\_\_ | Инициализация переменных класса, проверка на корректность заданного изображения |
| — | analyze | Поиск объектов путем использования компьютерного зрения, сбор статистики |
| Начальное время работы обработки изображения | print\_statistics | Вывод статистики изображения |
| Изображение, порядковый номер составной части | analyze\_cropped | Обработка изображения, вывод статистики |
| — | analyze\_cropped\_parallel | Распараллеливание обработки большого изображения |

Описание функций класса *«ImageAnalysisApp»* представлено в таблице 5.4.

*Таблица 5.4. Описание функций класса «ImageAnalysisApp»*

|  |  |  |
| --- | --- | --- |
| Входные данные | Имя | Описание |
| Объект интерфейса | \_\_init\_\_ | Инициализация переменных, настройка параметров интерфейса, запуск корректировки интерфейса |
| — | create\_interface | Корректировка интерфейса |
| — | select\_input\_folder | Выбор папки для загрузки изображений |
| — | select\_output\_folder | Выбор папки для отгрузки изображений |
| — | start\_threading | Запуск интерфейса в отдельном потоке |
| — | process\_images | Распараллеливание обработки изображений |
| Путь до изображения на устройстве | process\_image | Обработка изображения |

# **Рекомендации пользователю**

Выберете папку для загрузки изображений, а также папку для обработанных изображений. После выбора нажмите на кнопку *«Process Images».* Прогресс обработки изображений указывается в нижней части программы.

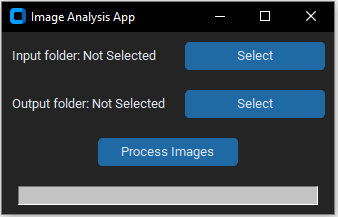
# **Рекомендации программисту**

Для запуска программы необходим Python, а также 64-битная операционная система Windows, или Linux, или macOS. Необходимые библиотеки: opencv-python версии 4.8.1.78, customtkinter версии 5.2.0. Для работы с кодом необходим PyCharm версии не ниже 2022.2.1.

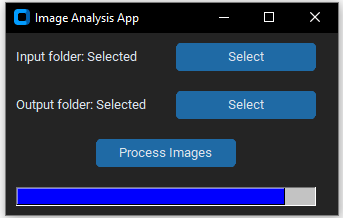
Минимальное необходимое место на диске: 0.5 МБ. Минимальное необходимое количество оперативной памяти: 500 МБ.

# **Контрольный пример**

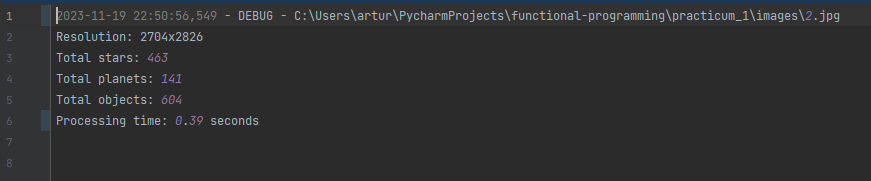
В данном разделе представлен контрольные пример, демонстрирующий способность выполнять комплексную параллельную обработку изображений.



*Рисунок 8.1. Интерфейс программы*



*Рисунок 8.2. Интерфейс программы во время обработки*



*Рисунок 8.3. Вывод статистики*

*Рисунок 8.4–8.5. Пример обработки изображения*

# **Заключение**

В рамках представленной работы был разработан и успешно реализован алгоритм для комплексной параллельной обработки, анализа и интерпретации астрономический изображений. Этот алгоритм направлен на изучение ключевых метрик, связанных с распознаванием космических объектов, таких как звёзды и планеты.