



Санкт-Петербургский государственный университет

Реализация библиотеки по обработке изображений с использованием GPU для платформы .NET

Савельева Полина Андреевна, группа 22.Б07-мм

Научный руководитель: к. ф.-м. н. С. В. Григорьев, доцент кафедры информатики

Санкт-Петербург
2023

- Использование GPU значительно ускоряет обработку изображений и снижает нагрузку на CPU
- Доступность графических процессоров позволяет применять их в сферах, где требуется обрабатывать большое количество данных (компьютерное зрение, медицина, графика и дизайн)
- Идея — предоставить инструмент, позволяющий эффективно использовать возможности GPU при работе с изображениями

- **Magick.NET** является .NET-оберткой для ImageMagick, позволяющая разработчикам использовать функциональность ImageMagick в приложениях, без необходимости вызова командной строки. Она поддерживает множество операций над изображениями, но только на CPU в отличие от ImageMagick.
- **ImageSharp**: библиотека помимо основных инструментов по работе с изображениями предоставляет своего рода строительные блоки, с помощью которых пользователи могут разрабатывать дополнительные функции. Не имеет встроенных функций для работы с GPU.
- **Emgu CV и opencvsharp** являются .NET-обёртками библиотеки OpenCV, поддерживающие обработку на GPU, но только видеокарт Nvidia.

Постановка задачи

Целью работы является реализация библиотеки по обработке изображений с использованием GPU

Задачи:

- Реализовать следующие возможности:
 - ▶ Сохранение в различных форматах
 - ▶ Изменение размера (resize)
 - ▶ Обрезка (crop)
 - ▶ Добавление водяного знака (watermark)
- Сравнить производительности текущей реализации и аналогов
- Оформить и опубликовать nuget пакет

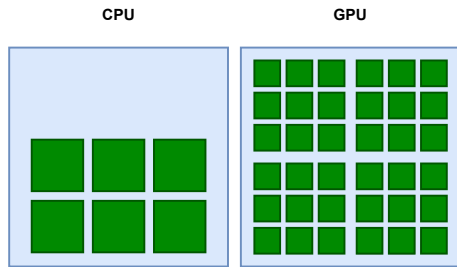


Рис.: Соотношение количества ядер на CPU и GPU

- Если в CPU могут быть 2–16 ядер, то в GPU их сотни и тысячи
- Наличие большого количества ядер обеспечивает параллелизм и как следствие высокую эффективность при обработке изображений — ускорение в 400 раз
- Brahma.FSharp — F# инструмент для работы с GPU различных производителей без использования низкоуровневых языков