

УДК 004.932:004.051

ПОДХОДЫ К ПОВЫШЕНИЮ ПРОИЗВОДИТЕЛЬНОСТИ ПРОГРАММЫ ОБРАБОТКИ ИЗОБРАЖЕНИЯ ФОРМАТА 360 ГРАДУСОВ

Рысьева И.А. (Университет ИТМО)

Научный руководитель – к.т.н. Балакшин П.В.
(Университет ИТМО)

Рассмотрен стандартный метод обработки изображений формата 360 градусов и проведен анализ реализации программных продуктов, использующих этот метод. Выявлены узкие места стандартной реализации и предложены методы повышения производительности алгоритма за счет введения многопоточности в существующую реализацию.

Введение. Современные программные средства обработки изображений формата 360 требуют больших вычислительных ресурсов или высоких трудозатрат со стороны пользователя. Достаточно часто коммерческое программное обеспечение осуществляет обработку изображений неоптимальными способами. Большинство коммерческих продуктов используют известную реализацию алгоритма из библиотек компьютерного зрения семейства OpenCV для преобразования изображений с двухобъективных камер.

По результатам анализа работы коммерческих программных продуктов Kolor Autopano и Samsung ActionDirector выявлено, что программы во время считывания файлов и преобразования изображений выполняются в однопоточном режиме. Такая реализация значительно замедляет работу пользователя при обработке большого количества данных, а также не использует в полной мере имеющиеся современные вычислительные ресурсы.

Цель исследования – произвести анализ работы алгоритма и оценить возможность повышения производительности алгоритма преобразования изображения с камер с двумя объективами «рыбий глаз» к формату эквидистантной панорамы.

Описание исследования. Для возможности подробного исследования алгоритма необходимо реализовать программу-аналог коммерческих приложений для преобразования изображений. Для преобразования изображения выбрана реализации библиотеки компьютерного BoofCV для языков JVM, которая является аналогом библиотеки OpenCV.

Для достоверности исследования был проведен опыт по преобразованию одинаковых изображений с помощью коммерческих ПО и разработанной программы на языке Kotlin. Полученные измерения показывают, что реализация программы для анализа имеет архитектуру аналогичную коммерческим реализациям. При анализе выявлено, что алгоритм осуществляет последовательную обработку изображения для каждой из линз камеры с объективом «Рыбий глаз», после чего осуществляет совмещение развернутых полупанорам. Таким образом для повышения производительности в данной реализации целесообразно вводить многопоточность в стандартную реализацию. При введении многопоточности обработка изображения с каждой из линз может осуществляться параллельно, и только последний этап совмещения изображений будет осуществляться в одном потоке.

Выводы. Для повышения производительности работы приложения могут быть использованы методы введения многопоточности на следующих этапах обработки изображения:

- считывание изображения;
- подготовка данных о камере;
- подготовка полупанорам.

Данный подход также может быть использован для многообъективных камерных систем.

Рысьева И.А. (автор)

Балакшин П.В. (научный руководитель)