

Оптимизация алгоритма ORB-SLAM2 для платформы с графической картой NVIDIA

Группа

7303

Студент

Державин Денис Павлович

*Руководитель к. т. н.
доцент*

Первицкий Александр Юрьевич

Консультант

*Чернокульский
Викторович*

Владимир

Цель и задачи

Цель: оптимизация алгоритма ORB-SLAM2 для платформы с графической картой NVIDIA.

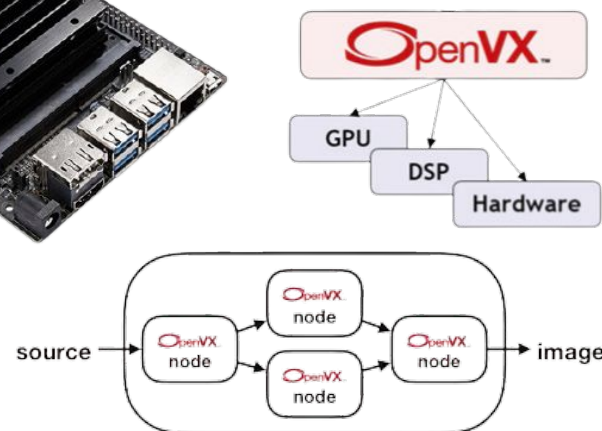
Актуальность: алгоритм ORB-SLAM2 является одним из самых надёжных и недорогих алгоритмов SLAM, но показывает высокие результаты только на мощных дорогих платформах.

Задачи:

- разработать оптимизированную версию ORB-SLAM2 для платформы с видеокартой NVIDIA;
- исследовать исходную и оптимизированную версии алгоритма;
- внедрить решение в ROS.

Используемые технологии

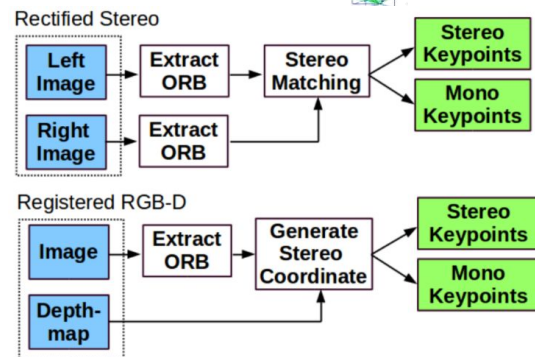
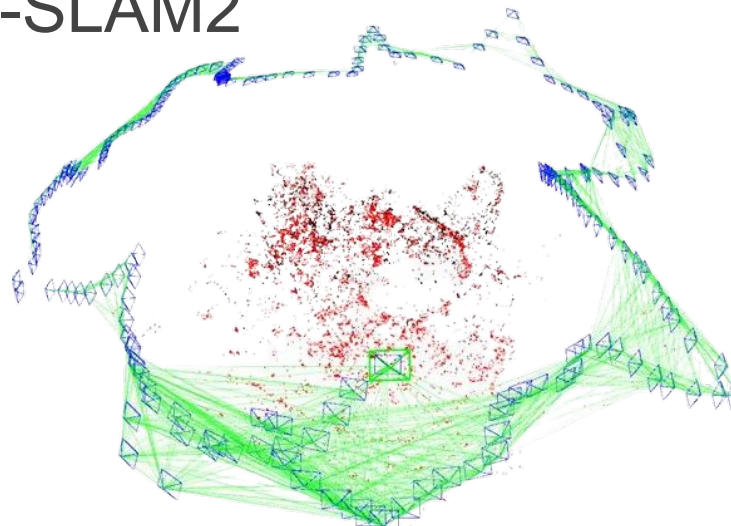
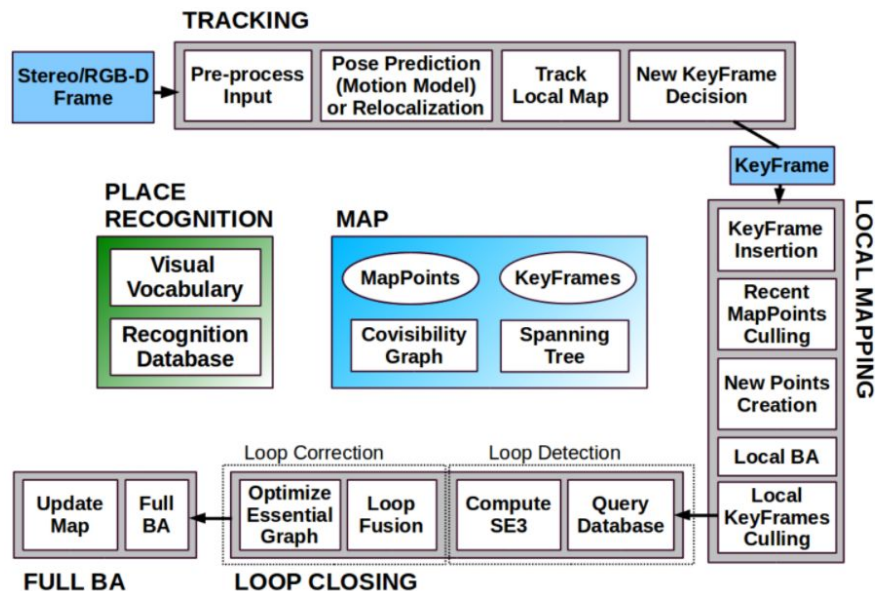
- **SLAM** – одновременная локализация и построение карты
- **OpenVX** – стандарт кроссплатформенного ускорения приложений компьютерного зрения
- **VisionWorks** – реализация стандарта OpenVX от компании NVIDIA для платформ своей линейки видеокарт
- **ROS** – универсальное программное обеспечение для создания приложений робототехники
- **Технология CUDA** – расширение языка C и ПО для написания кода для видеокарт компании NVIDIA



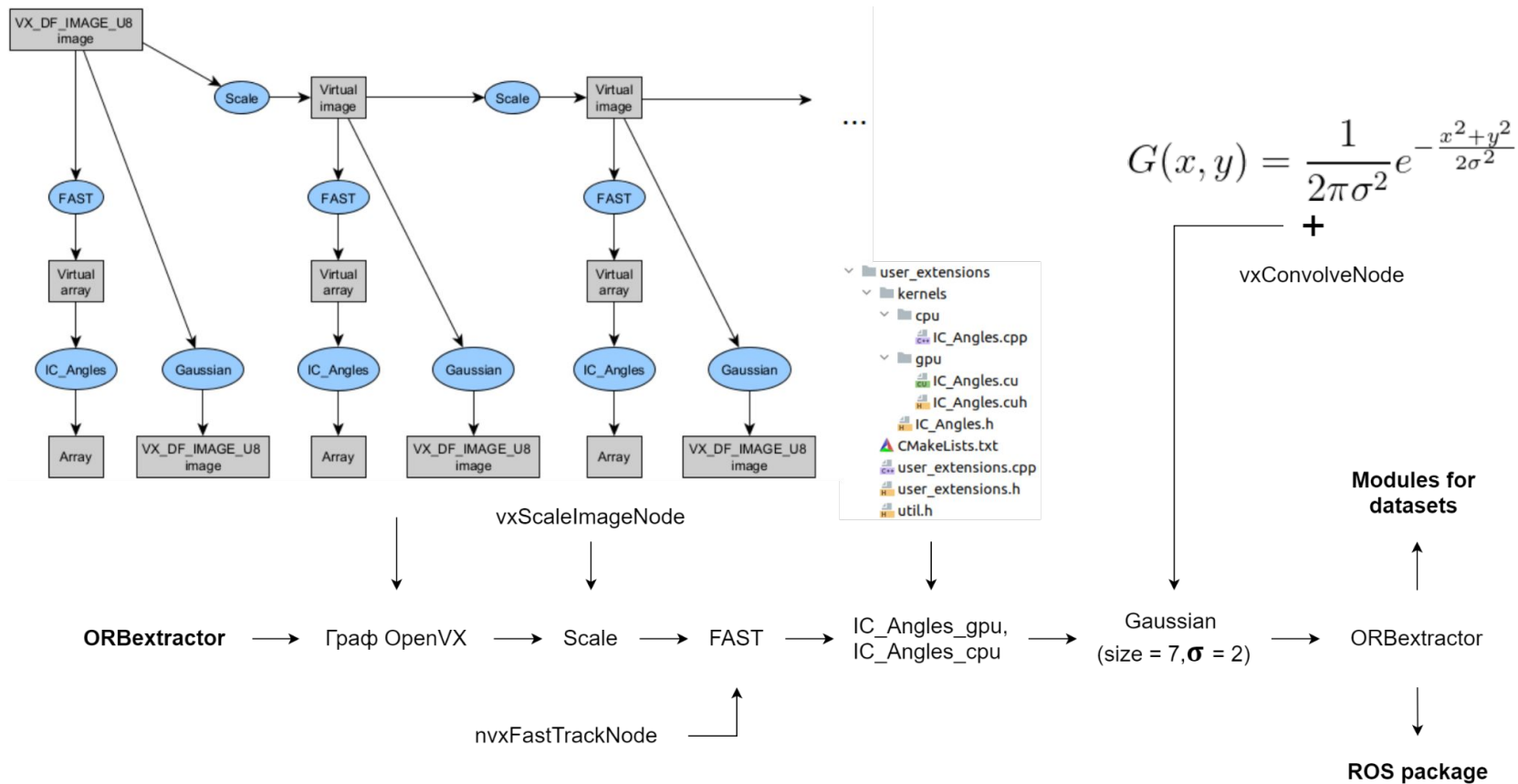
Обзор структуры алгоритма ORB-SLAM2

📝 Image Matching Using SIFT, SURF, BRIEF and ORB:
Performance Comparison for Distorted Images

20-40 мс, SURF – 30-80 мс, а SIFT – не менее 100 мс
на одной и той же машине

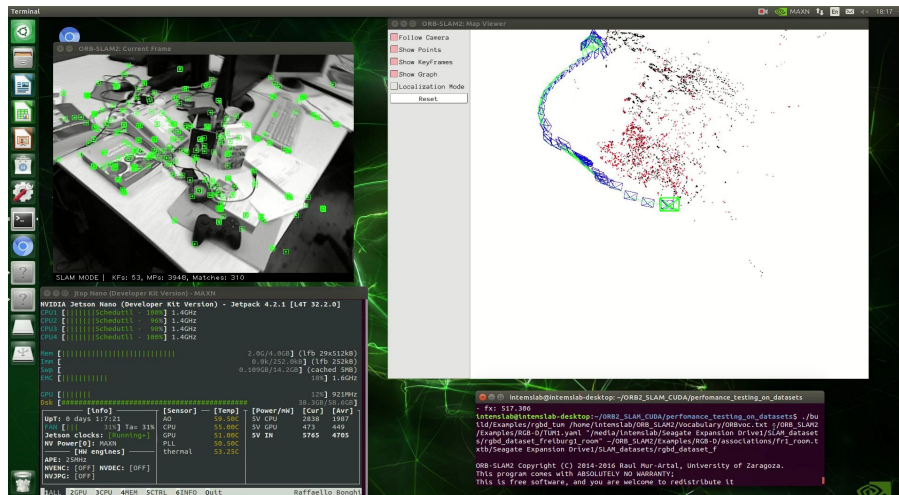


Описание оптимизации алгоритма ORB-SLAM2



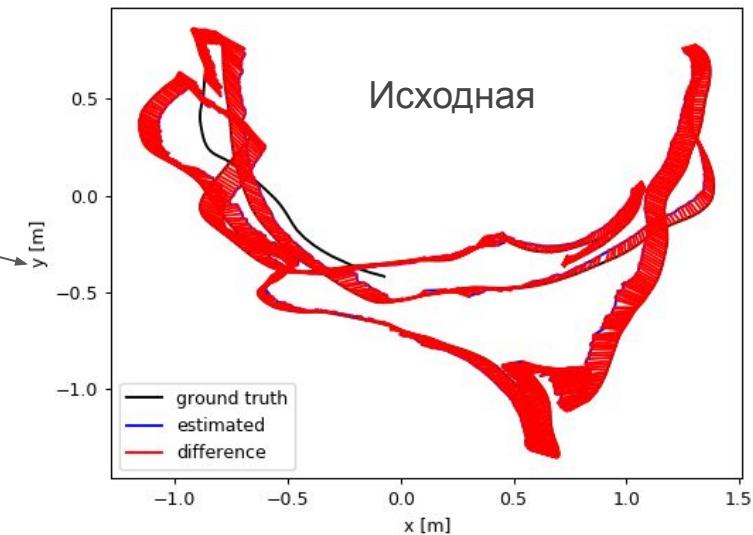
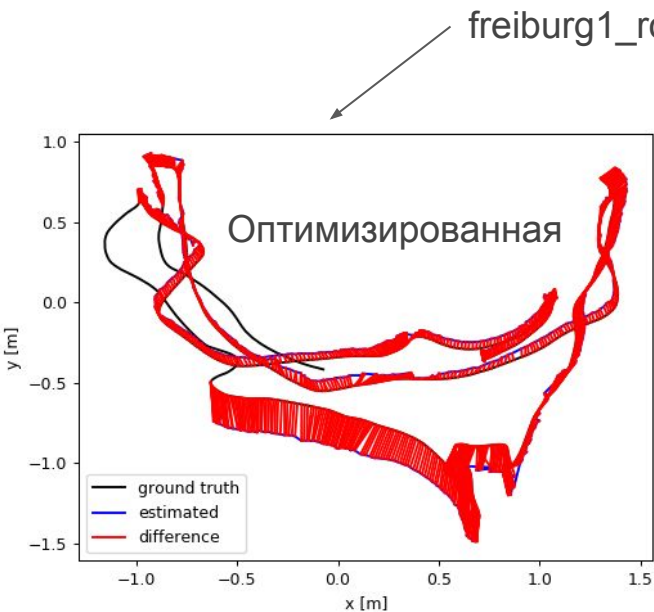
Тестирование решения

TUM RGB-D dataset and benchmark for visual SLAM evaluation



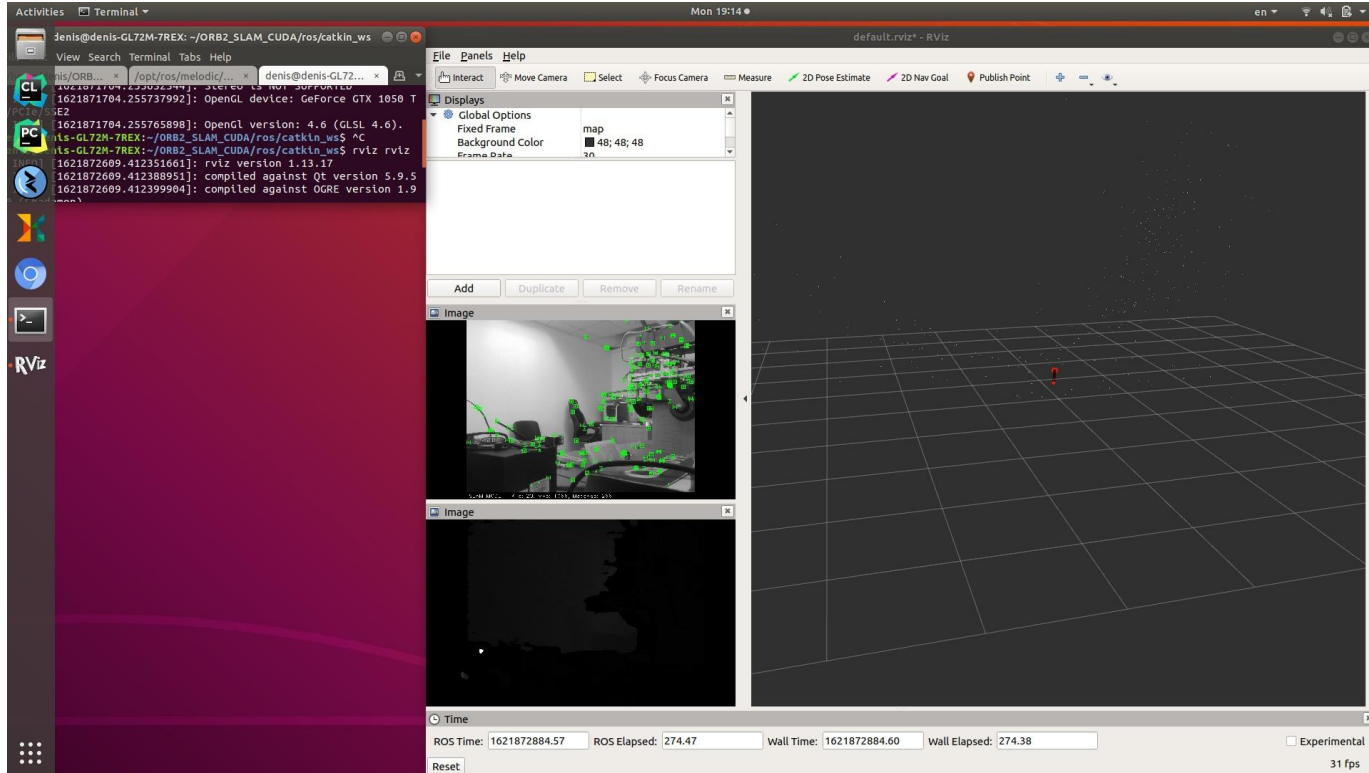
Датасет	Исходная версия		Оптимизированная версия		Ускорение	
	среднее, с	медиана, с	среднее, с	медиана, с	среднее, %	медиана, %
freiburg1_xyz	0,125680	0,124860	0,104828	0,103956	19,8	20,0
freiburg1_desk	0,119166	0,125679	0,108953	0,106623	9,3	17,8
freiburg1_room	0,124982	0,117748	0,0919278	0,0853231	35,9	28,0

Тестирование решения



Датасет	Исходная версия			Оптимизированная версия		
	среднее, м	медиана, м	стд. отклонение, м	среднее, м	медиана, м	стд. отклонение, м
freiburg1_xyz	0,011819	0,010418	0,006608	0,008602	0,007390	0,005652
freiburg1_desk	0,012410	0,010611	0,008392	0,013002	0,011024	0,009035
freiburg1_room	0,085198	0,089002	0,027212	0,091454	0,068979	0,051566

Внедрение в ROS



Апробация работы

Проект доступен на Github по ссылке: https://github.com/vchernokulsky/ORB2_SLAM_CUDA.git


Заключение

Результаты:

- ✓ Была разработана оптимизированная версия алгоритма;
- ✓ Оптимизированная и исходная версии были исследованы, исследование показало, что оптимизированная версия работает в среднем в 1,2 раза быстрее, чем исходная;
- ✓ Результат был внедрён в ROS.

Дальнейшее развитие проекта предполагает:

- повышение точности построения карты оптимизированной версией алгоритма;
- переход на стандарт OpenVX 1.3 и собственная реализация узлов для исполнения на видеокарте;
- доработку бэкенда графа вычислений с целью создания версии, портируемой на другие платформы, например, платформы, использующие OpenCL для исполнения кода на графическом процессоре;
- исследование и оптимизация других участков алгоритма;
- исследование алгоритма ORB-SLAM3 и возможности внедрения в него текущих улучшений.



Спасибо за внимание