

Безреференсная метрика для оценки качества траектории из облаков точек

- **Отчёт по производственной практике**

Салью Артур Кристофович
Группа 19.Б11-мм

Научный руководитель

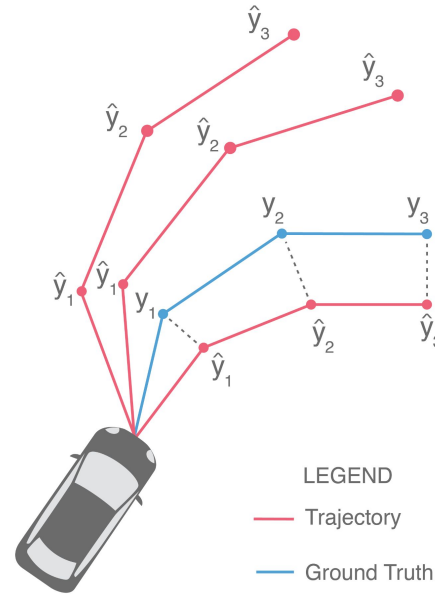
ст.пр. Я. А. Кириленко

Консультант

Инженер-исследователь, Сколтех, А. В. Корнилова

Траектории автономной системы

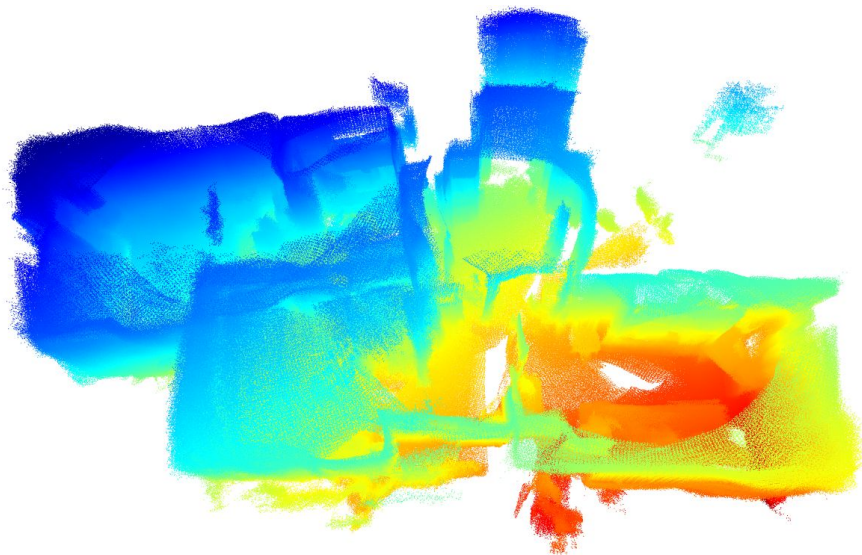
- Одометрия — оценка положения системы в пространстве
- Референсные данные
 - *GNSS, MoCap*
- Ограничения
 - Погрешность
 - Особенности сцены
 - Дороговизна



Источник: <http://ai.stanford.edu/blog/trajectory-forecasting/>

Безреференсные метрики (1/2)

- ◎ Артефакт работы сенсоров — облако точек
- ◎ Особенности карты облаков:
 - Шум
 - Консистентность



Карта, собранная из набора облаков точек

Безреференсные метрики (2/2)

Базовые безреференсные метрики:

- ◎ MME^[1] — Mean Map Entropy
- ◎ MPV^[2] — Mean Plane Variance
- ◎ Mutually Orthogonal Metric (MOM)^[3]
 - Точки на ортогональных плоскостях

[1] [J. Razlaw et al. “Evaluation of registration methods for sparse 3D laser scans”, ECMR 2015](#)

[2] [D. Droschel et al. “Local multi-resolution representation for 6D motion estimation and mapping with a continuously rotating 3D laser scanner”, ICRA 2014](#)

[3] [A. Kornilova et al. “Be your own Benchmark: No-Reference Trajectory Metric on Registered Point Clouds”, ECMR 2021](#)

Библиотека map-metrics

map-metrics^[4] — библиотека с безреференсными метриками

- ⊙ Написана на Python
- ⊙ Апробирована только на LiDAR данных

Недостатки текущей реализации

- ⊙ Интеграция в ROS, CloudCompare^[5]
- ⊙ Низкая производительность
- ⊙ Невозможность задания пользовательских параметров

[4] <https://github.com/MobileRoboticsSkoltech/map-metrics>

[5] <https://www.cloudcompare.org/>

Цель и задачи

Цель работы — реализовать кроссплатформенную библиотеку безреференсных метрик на языке C++ с Python интерфейсом.

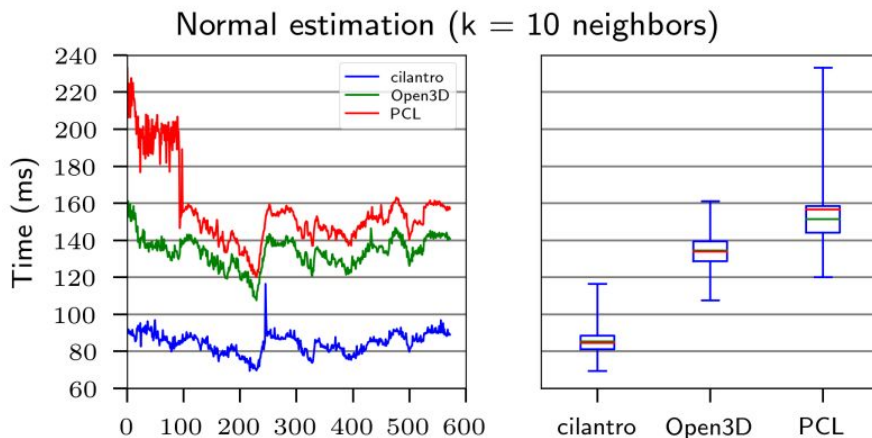
Задачи

- ◎ Реализовать библиотеку метрик на языке программирования C++;
- ◎ Поддержать Python обвязку и CI/CD Python пакета в индекс PyPI;
- ◎ Добавить возможность задавать параметры алгоритмов в зависимости от сценария использования датчиков;
- ◎ Оценить производительность и качество реализованных алгоритмов.

Реализация базовых метрик

Сравнение библиотек для работы с облаками точек

	Параллелизм	Зависимости
cilantro ^[6]	+	C++11, Eigen 3.3
Open3D ^[7]	+	C++14
PCL ^[8]	+	Boost 1.47, Eigen 3.0, FLANN 1.7.1



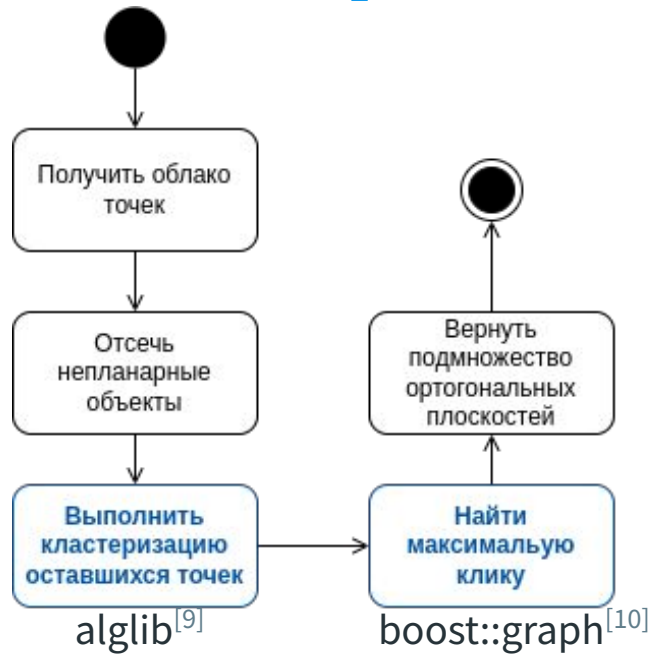
Источник: [K. Zampogiannis et al. “cilantro: A Lean, Versatile, and Efficient Library for Point Cloud Data Processing”, ACM 2018](#)

[6] <https://github.com/kzampog/cilantro>

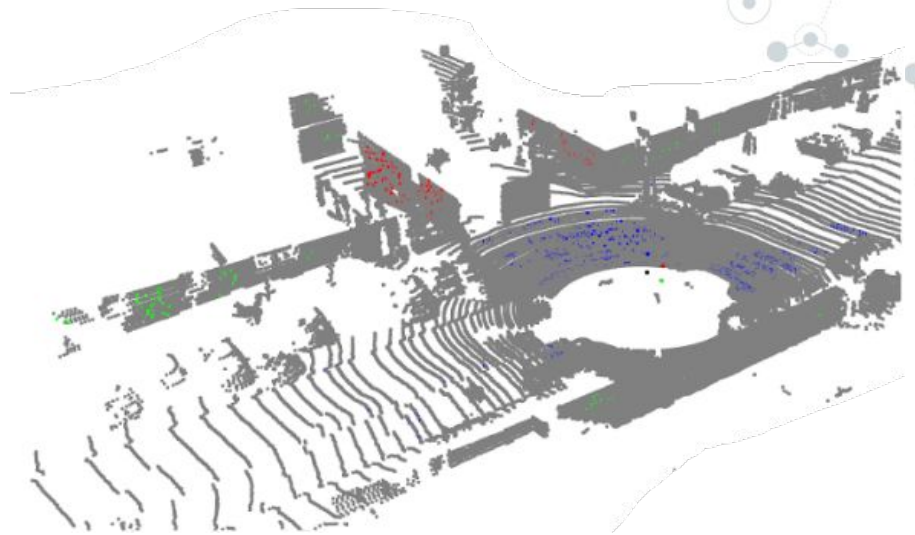
[7] <http://www.open3d.org/>

[8] <https://pointclouds.org/>

Извлечение ортогональных плоскостей



Этапы алгоритма извлечения ортогональных плоскостей



[9] <https://www.alglib.net/>

[10] https://www.boost.org/doc/libs/1_79_0/libs/graph/doc/index.html

[11] [A. Geiger et al. "Are we ready for autonomous driving? The KITTI vision benchmark suite", CVPR 2012](#)

Python пакет

Создание Python обвязки — pybind11^[12]

Сборка пакета

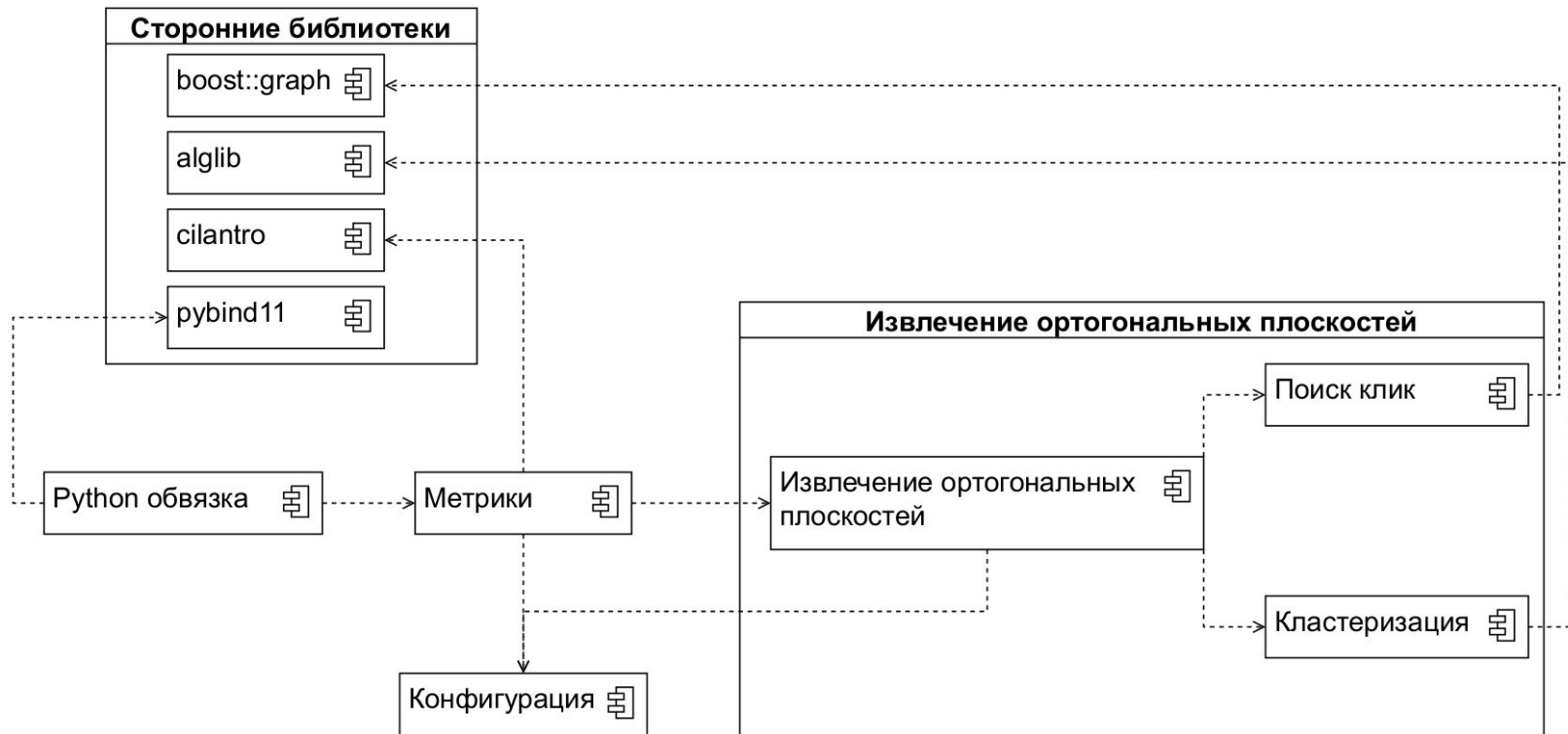
- ◎ Распространение в качестве колеса (Python wheel)
- ◎ Поддержка Centos 6, OS X 10.14+, Windows 7+

CI/CD

- ◎ GitHub Actions
- ◎ Автоматическая сборка и загрузка в PyPI

[12] <https://github.com/pybind/pybind11>

Диаграмма компонентов



Описание экспериментов

Данные

- ◎ 28 облаков, 100 тысяч точек

Оценка времени работы метрик

- ◎ 80 запусков, *time.process_time_ns()*

Оценка точности метрик

- ◎ Абсолютная погрешность
- ◎ Карта из 4722 облаков

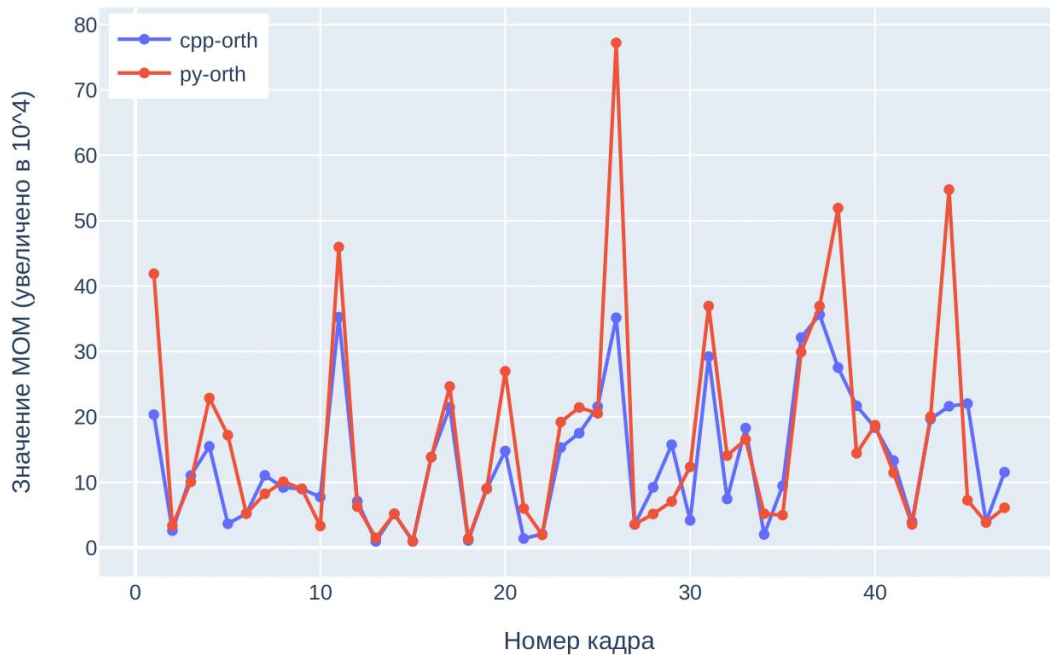
Сравнение времени работы

Алгоритм	Версия	Среднее арифметическое (с)	CI ($\alpha = 0.05$)
MPV	Python	24.48	[24.41, 24.55]
	C++	5.25	[5.22, 5.28]
MME	Python	19.27	[19.24, 19.31]
	C++	5.27	[5.25, 5.29]
Извлечение ортогональных плоскостей	Python	0.368	[0.365, 0.370]
	C++	0.038	[0.038, 0.039]

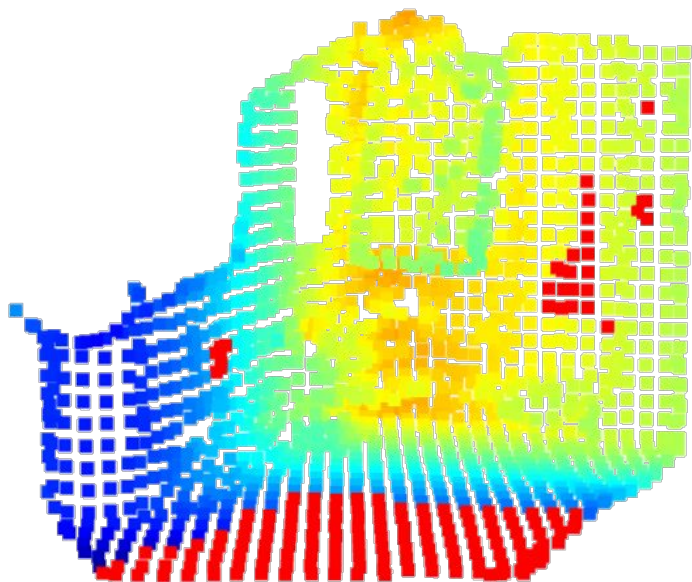
Сравнение точности

Абсолютная разница значений метрик

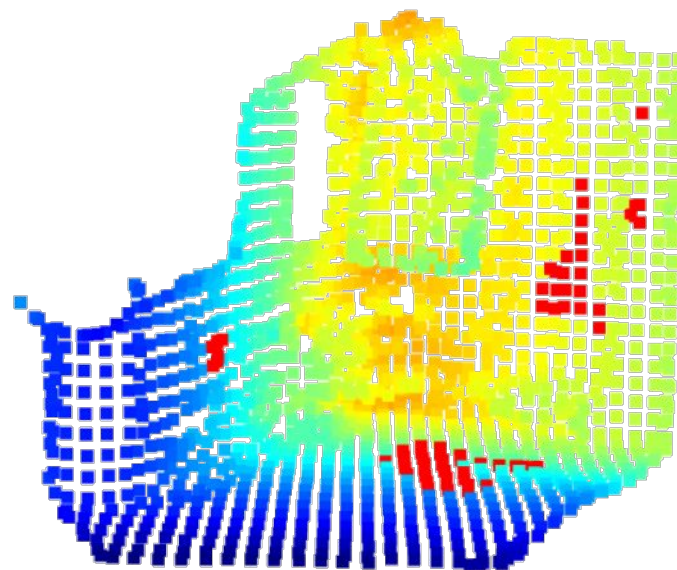
ΔMME	ΔMPV	ΔMOM
$6 \cdot 10^{-14}$	10^{-18}	$2 \cdot 10^{-18}$



Извлеченные плоскости



C++ версия



Python версия

Результаты

- ⦿ Реализована библиотека map-metrics на языке C++;
- ⦿ Поддержана Python обвязка и настроен CI/CD для публикации pip-пакета;
- ⦿ Добавлена возможность задания параметров алгоритмов для учёта сценария использования датчиков;
- ⦿ Проведена оценка качества и времени работы метрик.

Проект на GitHub:

<https://github.com/MobileRoboticsSkoltech/map-metrics>

Библиотека на PyPI:

<https://pypi.org/project/map-metrics/>