Курс «Трёхмерное компьютерное зрение»

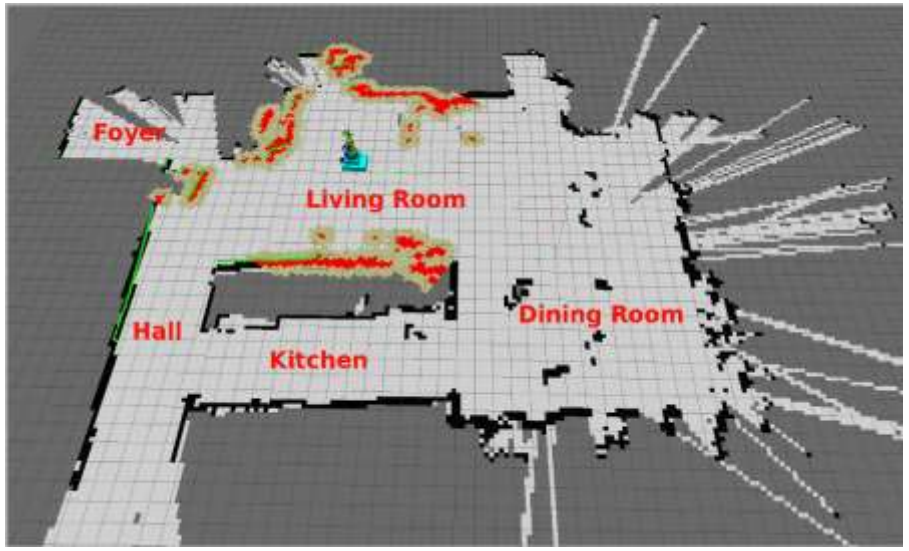
Тема №4B

«SLAM»

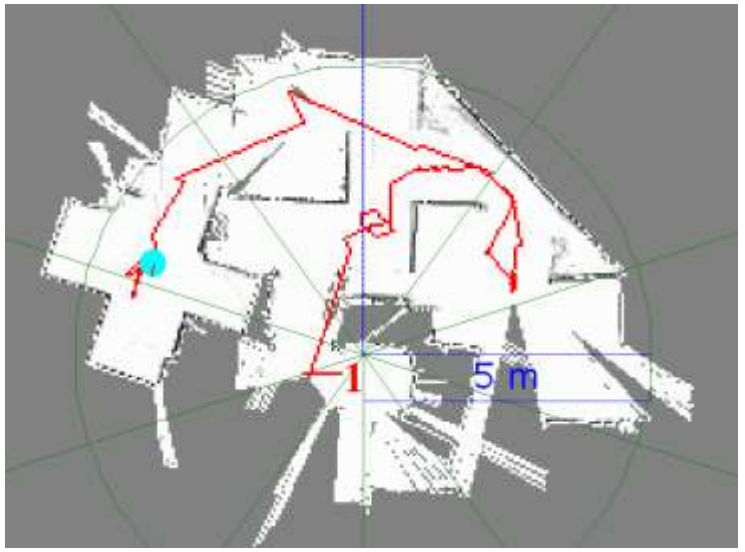
Антон Конушин

Simultaneous Localization and Mapping (SLAM)

Комбинация Одометрии (оценки траектории движения камеры), Картографирования (построения карты сцены), Локализации (оценки положения по карте), Закрытия циклов / Loop Closure



Карта какой-то квартиры



Карта и траектория

Если по данным камеры, тогда говорят Visual SLAM. Но могут использоваться и другие сенсоры (IMU, 2D Lidar, 3D Lidar, и т.д.)

SfM vs. SLAM

SfM

- Input is unordered set of images
- Focus is on precision, with aim to produce a good 3D model
- Offline, one-time process

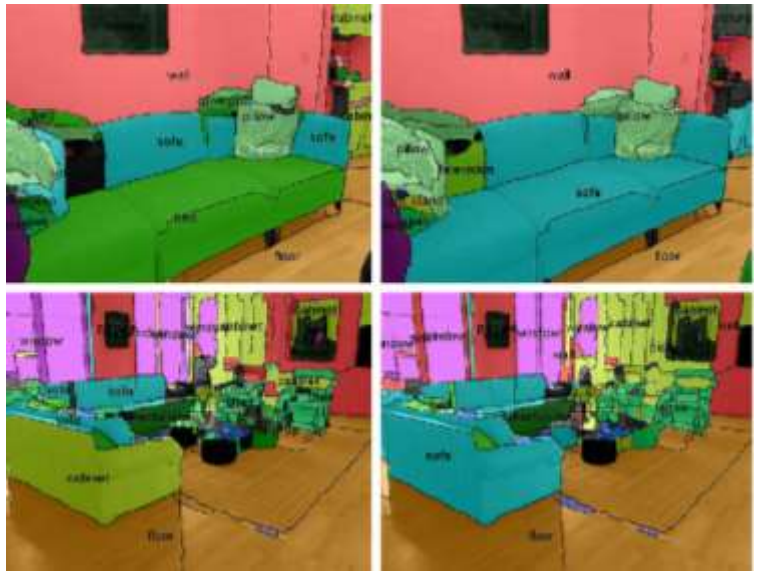
- Published mainly in vision conferences
- 3 papers with more than 1000 citations
- Complicated

SLAM

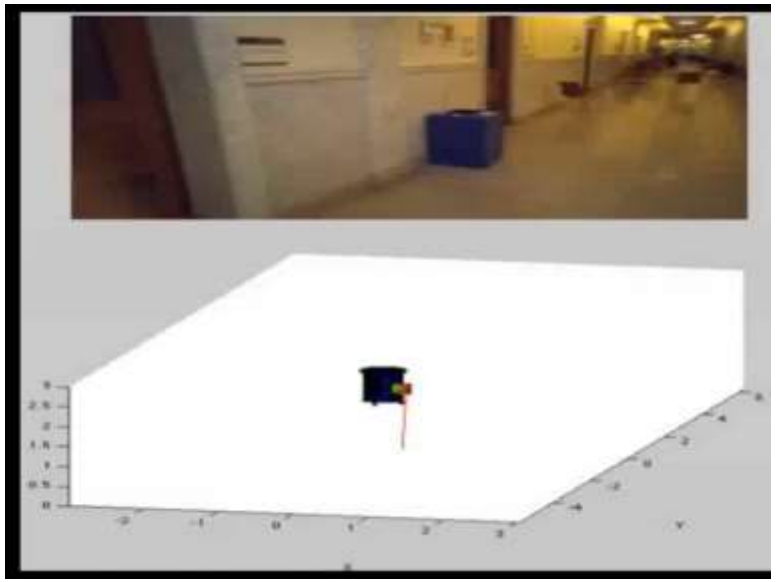
- Input is stream of images, stereo, or depth and sometimes IMU
- Focus is on speed and robustness, with aim to localize camera or robot
- Online process, possibly with relocalization
- Published mainly in robotics conferences
- 8 papers with more than 1000 citations
- Very complicated

Semantic SLAM

- Semantic SLAM или Total Scene Understanding
  - Одновременное распознавание сцены и SLAM, что дает полный ответ на вопрос «что где расположено» про всю сцену

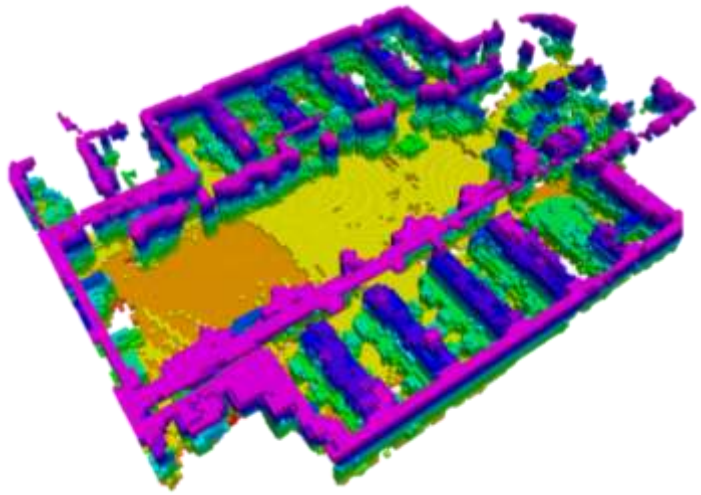


Semantic segmentation



Example of semantic SLAM

Source: https://natanaso.github.io

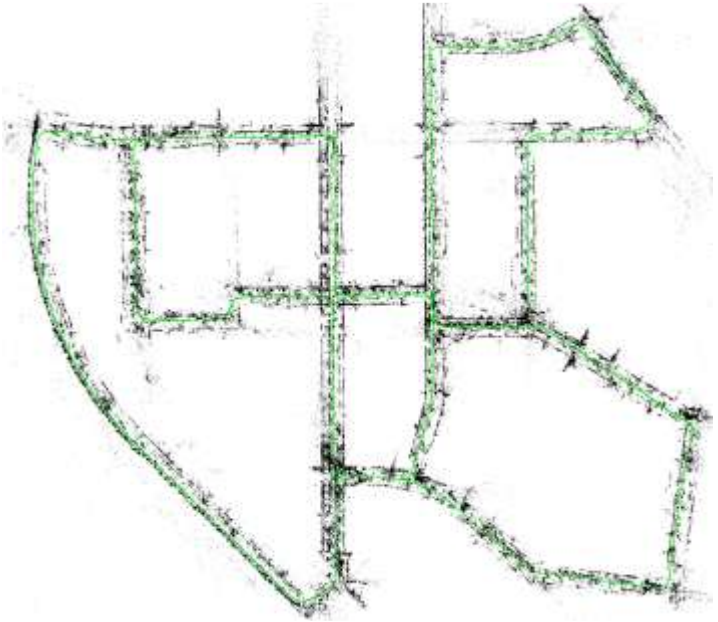


Volumetric semantic map

Source: https://arxiv.org/pdf/1801.07380.pdf

Оценка качества

«Карта» обычно представлена в виде набора ключевых кадров с позами и облаком точек. Непонятно, как оценивать точности таких карт. Поэтому качество SLAM обычно оценивают по траекториям

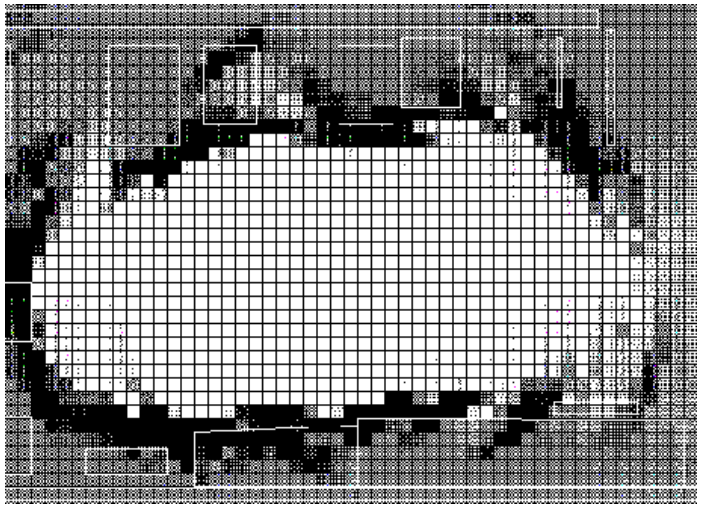


Sparse point cloud from monocular RGB images

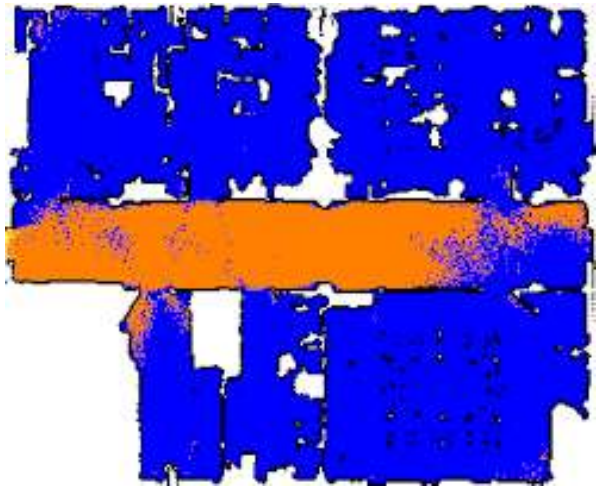
Dense point cloud from RGBD video

Семантические карты

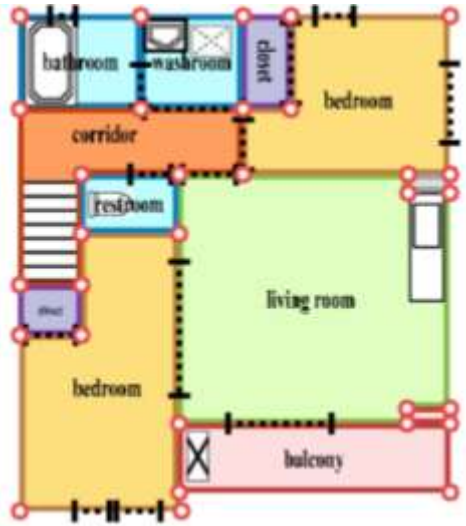
- Если карты строятся, то обычно они строятся в виде 2Д изображений.
- Для оценки точности требуется эталонная разметка этих карт



Occupancy grid map



Occupancy grid map with semantic labels

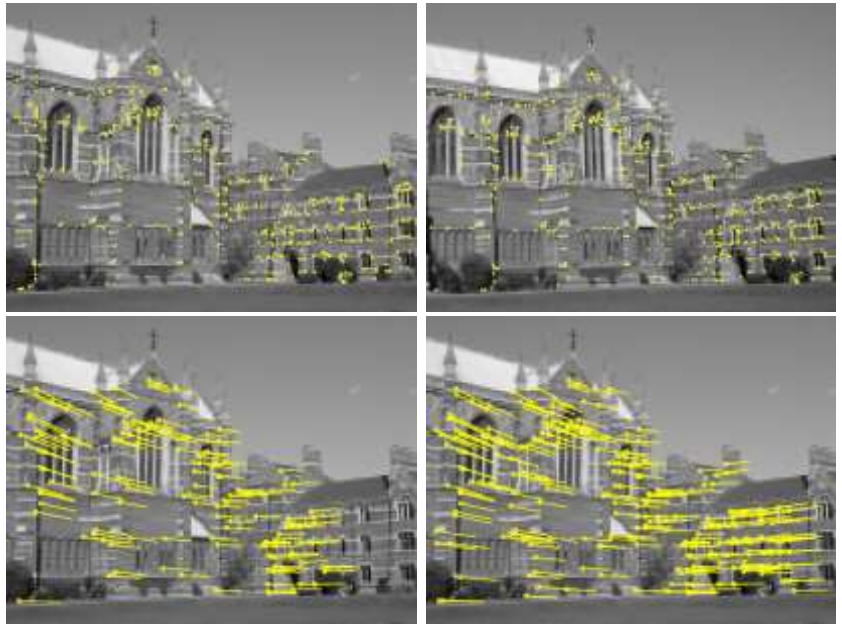


Vector map with labels

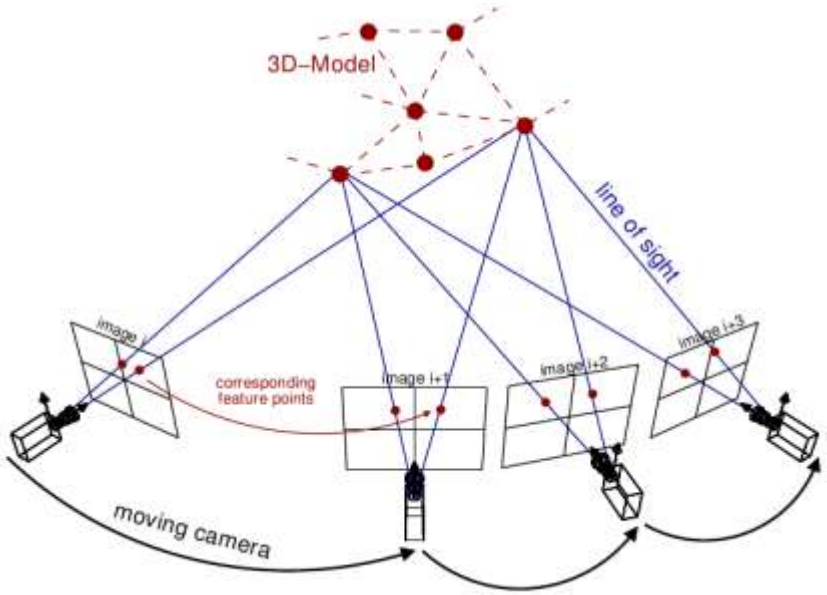
Классический SLAM

Классический Visual SLAM

- Классическими можно назвать методы, опирающиеся на точечные соответствия между изображениями, многовидовую геометрию и уточнение градиентной оптимизацией
- Фокус на последовательную обработку кадров и работу в реальном времени



Точки и их сопоставление



Локализация камеры и структура из движения

ORBSLAM

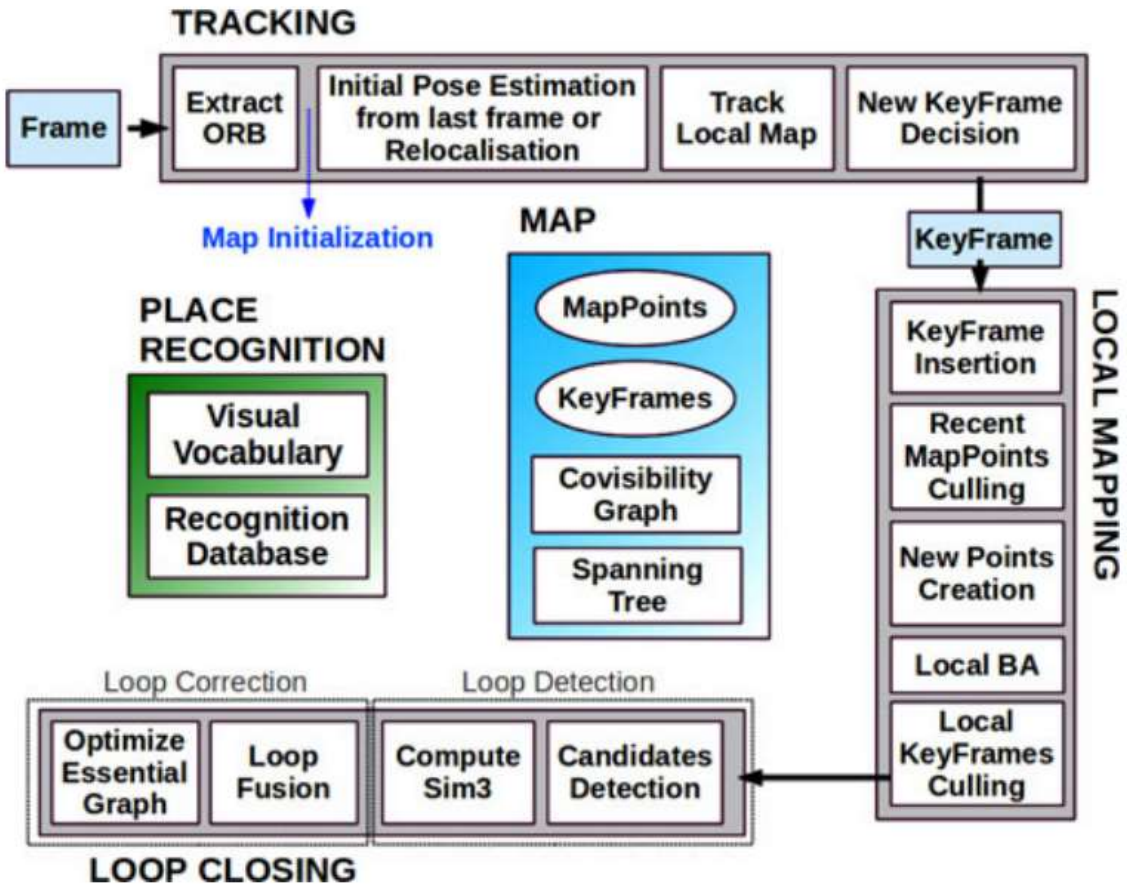


Fig. 1. ORB-SLAM system overview, showing all the steps performed by the tracking, local mapping, and loop closing threads. The main components of the place recognition module and the map are also shown.

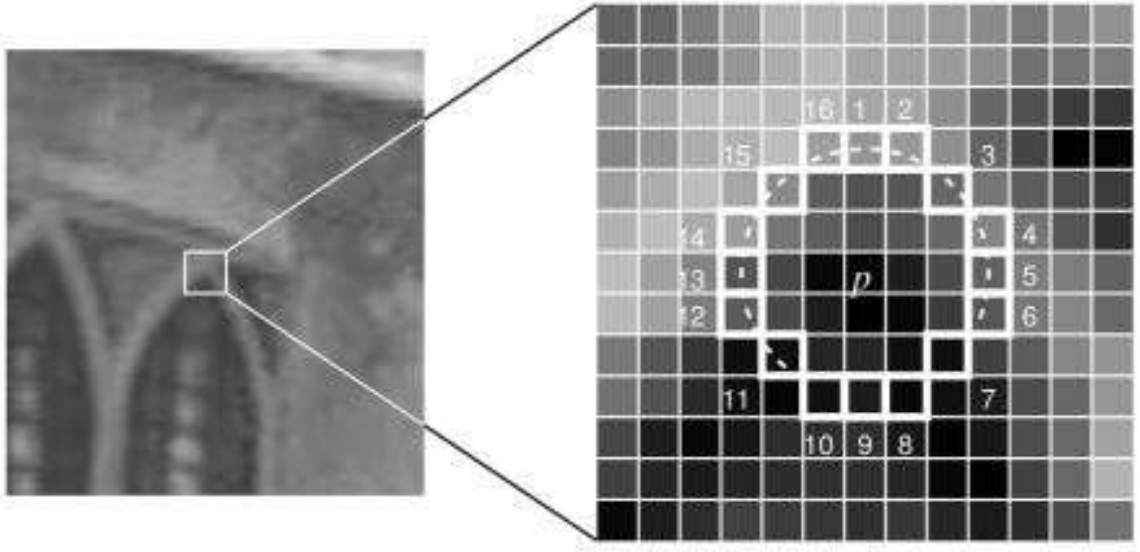
- Первая работа цикла открытых реализацией SLAM-методов
  - https://github.com/raulmur/ORB_SLAM
- Опирается на быстрый детектор ORB
- Карта сцены описывается набором ключевых кадров и набором 3д точек, наблюдаемых на этих кадрах
- 3 нити исполнения (Tracking, Local Mapping, Loop Closing), работающие параллельно
- Tracking – отслеживание точек и оценка позы камеры в локальной карте
- Local Mapping – добавление нового ключевого кадра в карту и последствия этого
- Loop Closure – определение циклов и уточнение карты и траектории из-за этого

R.Mur-Artal, J.M.M.Montiel, J.D. Tardos. ORB-SLAM: A Versatile and Accurate Monocular SLAM System. IEEE Transactions on robotics, 2015

ORB (Oriented FAST and Rotated BRIEF)

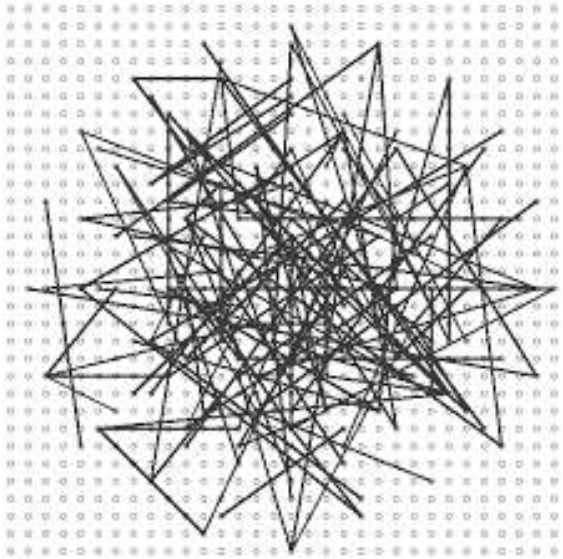
«Вершина» необучаемых аппроксимаций SIFT

Идея FAST



- Выбираем 16 пикселов на дискретной окружности (алгоритм Брезенхейма) вокруг точки p
- Углом будет считаться точка, если 12 последовательных пикселов либо светлее p больше, чем на порог t, либо темнее

Идея BRIEF



- Выбираем N случайных пар пикселов (a_i, b_i) в окрестности
- Каждой паре сопоставляем 1 (если яркость $I(a_i) > I(b_i)$) и 0, если наоборот
- Получаем бинарный код desc(p), и можем сравнить по расстоянию Хэмминга с другими

https://medium.com/data-breach/introduction-to-orb-oriented-fast-and-rotated-brief-4220e8ec40cf

ORBSLAM - Tracking

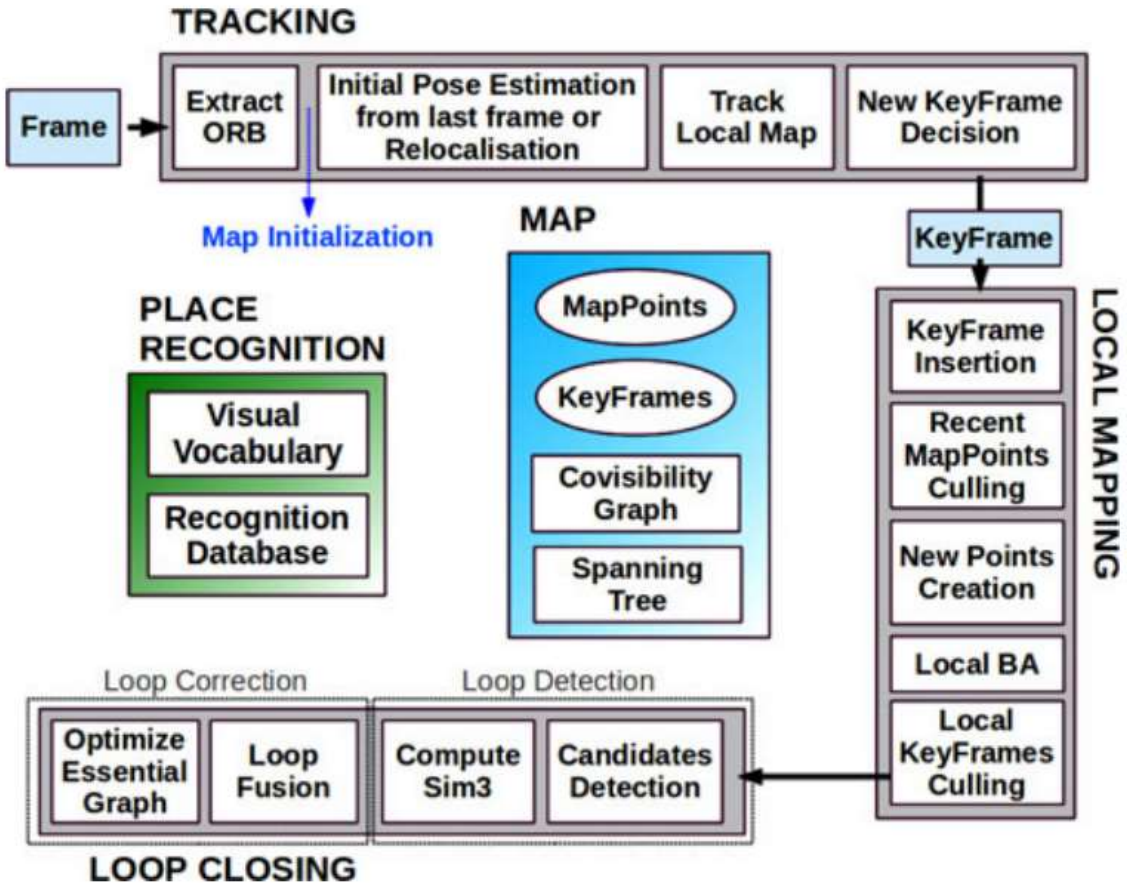
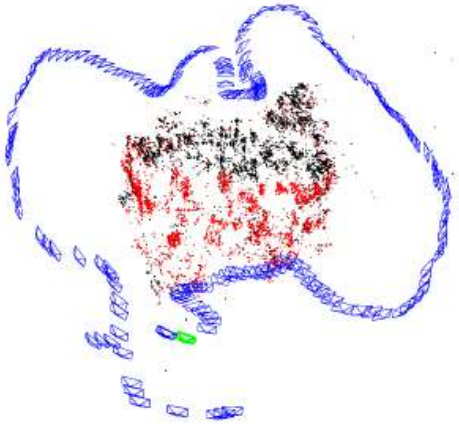


Fig. 1. ORB-SLAM system overview, showing all the steps performed by the tracking, local mapping, and loop closing threads. The main components of the place recognition module and the map are also shown.

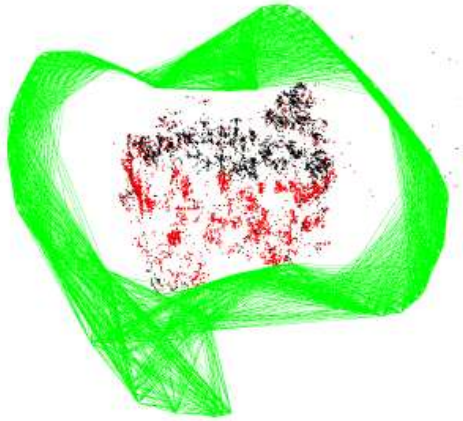
- На новом кадре находятся ORB точки
- Они сопоставляются с предыдущим кадром, и оценивается поза камеры.
- Если с предыдущим кадром не удается сопоставится, тогда релокализация
- Поскольку ORB ненадёжный, то точек сопоставляется мало, поэтому отдельный этап Track Local Map
- С предыдущих ключевых кадров видимые 3Д точки сцены проецируются на текущий кадр, фильтруются по эвристикам, и затем матчатся с несматченными ORB точками
- По итогу поза уточняет, и принимается решение, добавлять или нет новый ключевой кадр в карту

R.Mur-Artal, J.M.M.Montiel, J.D. Tardos. ORB-SLAM: A Versatile and Accurate Monocular SLAM System. IEEE Transactions on robotics, 2015

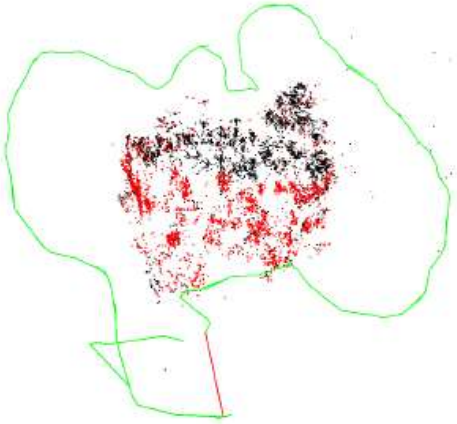
ORBSLAM - Mapping



(a) KeyFrames (blue), Current Camera (green), MapPoints (black, red), Current Local MapPoints (red)



(b) Covisibility Graph



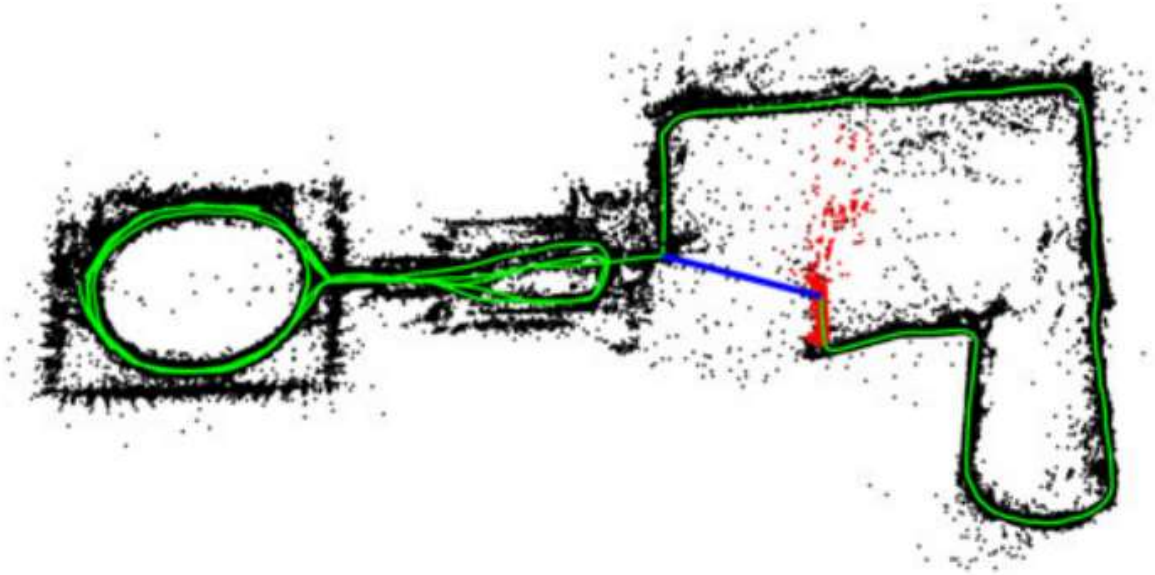
(c) Spanning Tree (green) and Loop Closure (red)

(d) Essential Graph

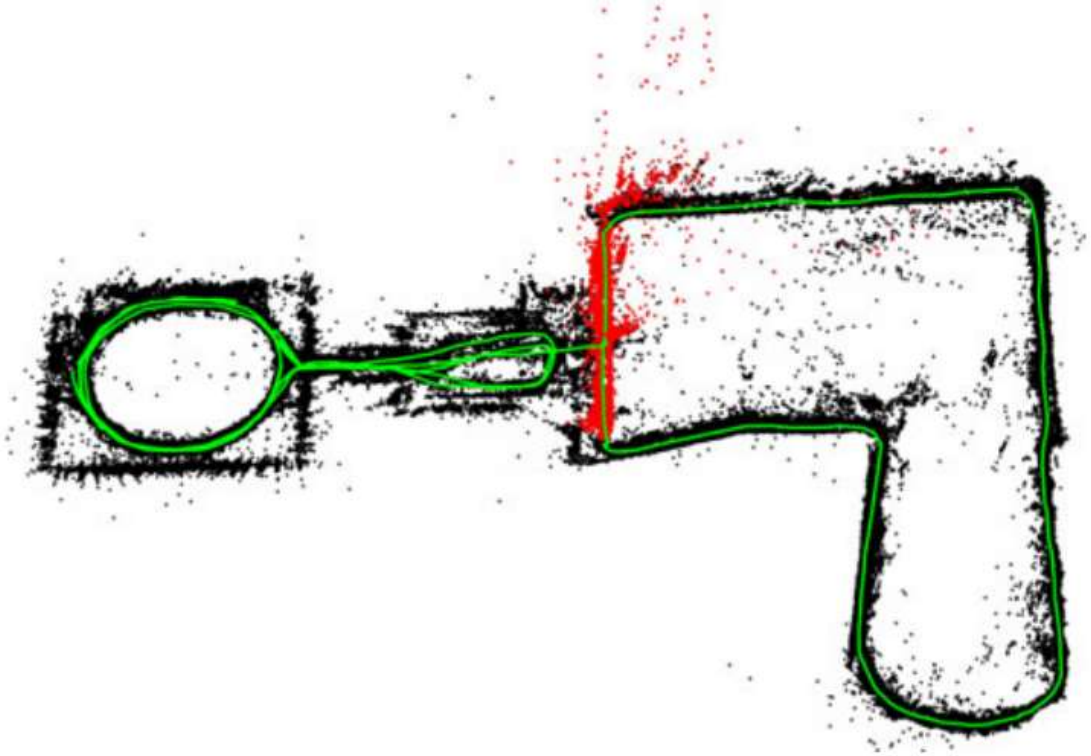
- Полная карта сцены – Co-visibility Graph
  - узлы это ключевые кадры
  - ребра связывают кадры, которые видят >= 15 одних и тех же точек
- Essential Graph – упрощённая карта сцены с прореженными связями, минимальное оставное дерево
- EG используется, когда нужно уточнить всю карту при закрытии циклов
- Ключевые кадры выбираются часто, предусмотрен механизм отбраковки лишних ключевых кадров
- Поиск похожих изображений по тем же ключевым точкам используется для детекции циклов и ре-локализации
  - https://github.com/dorian3d/DBoW2

ORBSLAM - Loop Closure

Закрытие цикла приводит к «скачкам» камеры и пересчёту карты

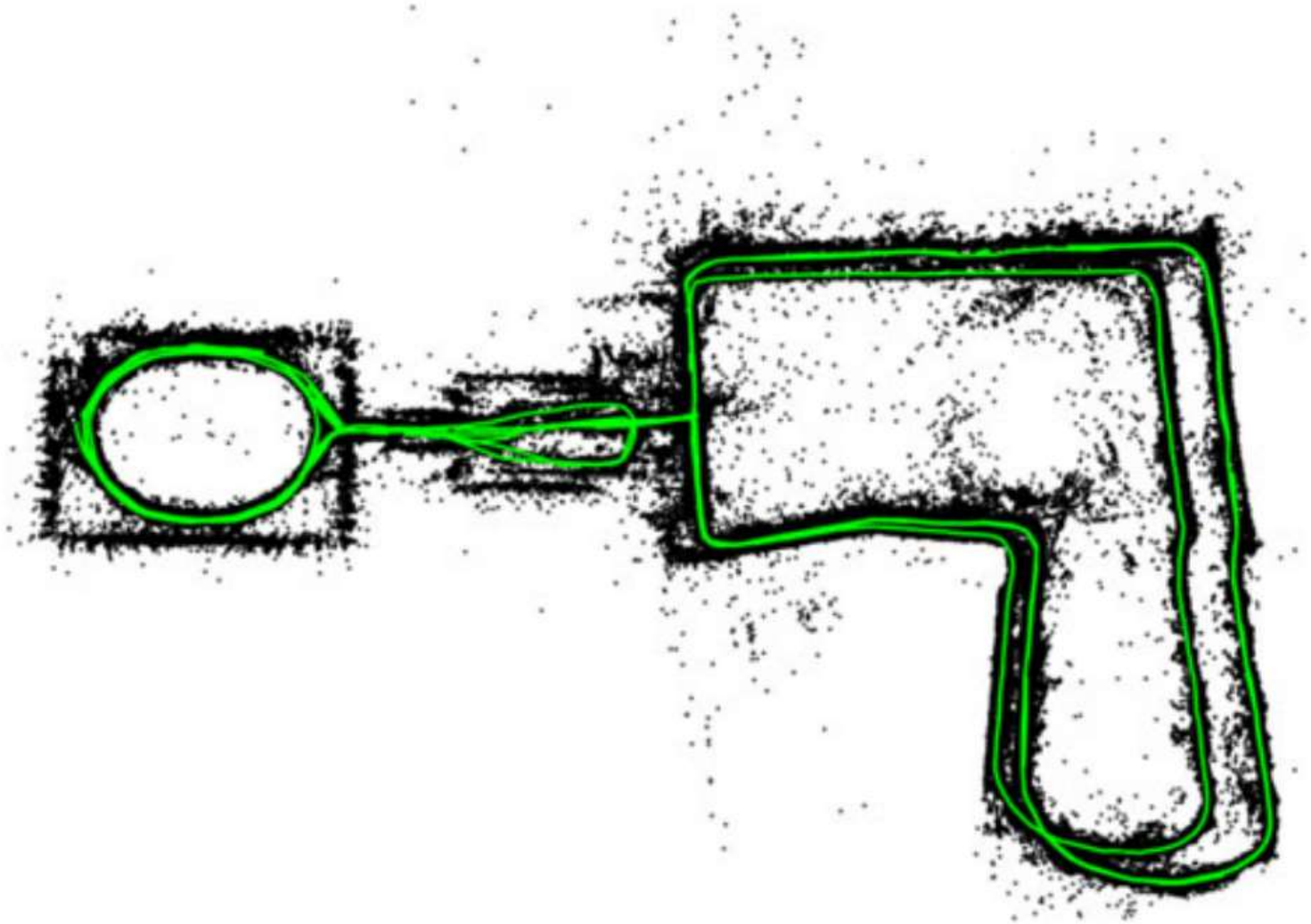


После закрытия цикла



До закрытия цикла

ORBSLAM - Ограничения закрытия циклов



Движение было в одном и другом направлении, поэтому картинки не были похожими, цикл не был обнаружен, и закрытие цикла не произошло

Скорости работы

TABLE I
TRACKING AND MAPPING TIMES IN NEWCOLLEGE

Thread	Operation	Median (ms)	Mean (ms)	Std (ms)
TRACKING	ORB extraction	11.10	11.42	1.61
	Initial Pose Est.	3.38	3.45	0.99
	Track Local Map	14.84	16.01	9.98
	Total	30.57	31.60	10.39
LOCAL MAPPING	KeyFrame Insertion	10.29	11.88	5.03
	Map Point Culling	0.10	3.18	6.70
	Map Point Creation	66.79	72.96	31.48
	Local BA	296.08	360.41	171.11
	KeyFrame Culling	8.07	15.79	18.98
	Total	383.59	464.27	217.89

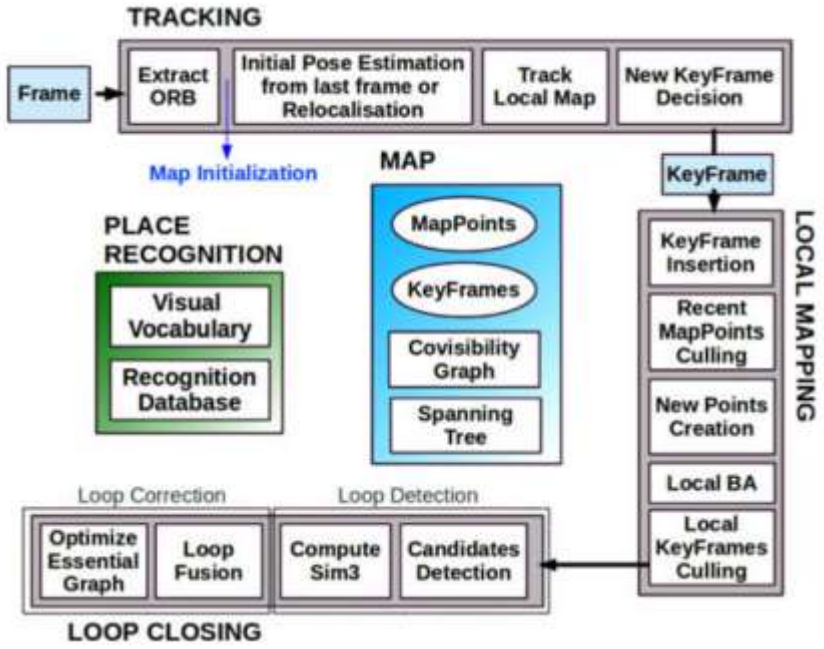
Скорости работы

TABLE II
LOOP CLOSING TIMES IN NEWCOLLEGE

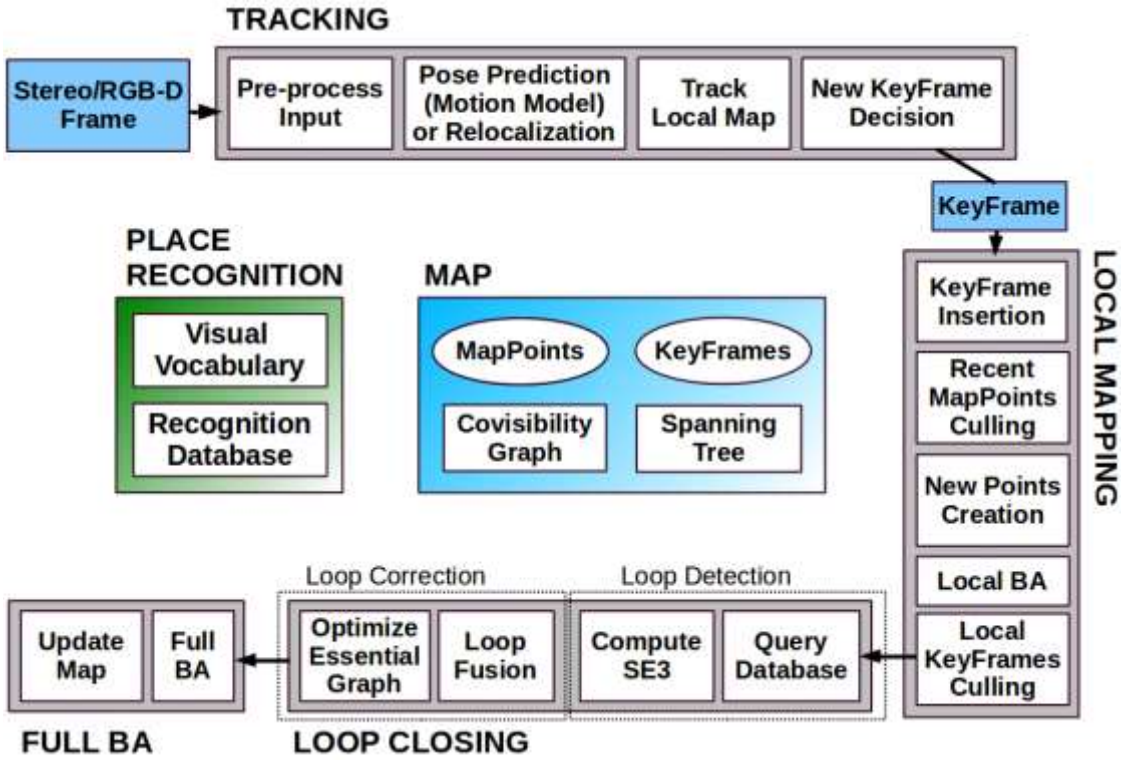
			Loop Detection (ms)		Loop Correction (s)		
Loop	KeyFrames	Essential Graph Edges	Candidates Detection	Similarity Transformation	Fusion	Essential Graph Optimization	Total (s)
1	287	1347	4.71	20.77	0.20	0.26	0.51
2	1082	5950	4.14	17.98	0.39	1.06	1.52
3	1279	7128	9.82	31.29	0.95	1.26	2.27
4	2648	12547	12.37	30.36	0.97	2.30	3.33
5	3150	16033	14.71	41.28	1.73	2.80	4.60
6	4496	21797	13.52	48.68	0.97	3.62	4.69

ORBSLAM2

Развитие ORBSLAM с добавлением уточнения всей карты (глобального BA) и работой со стерео-данными и RGB-D



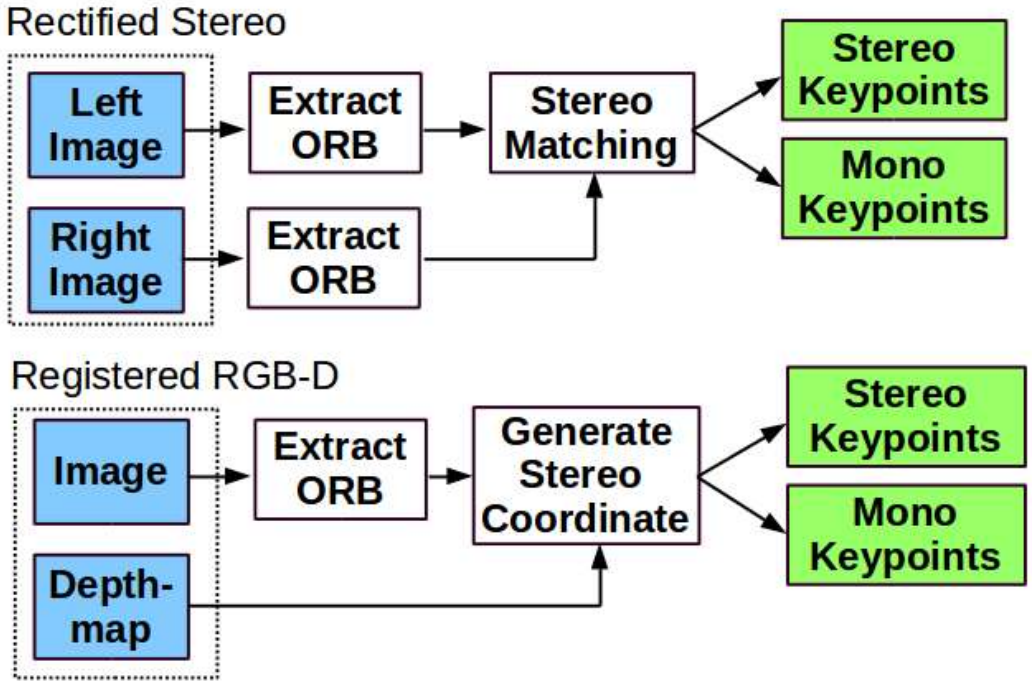
ORBSLAM



ORBSLAM2

Raúl Mur-Artal and Juan D. Tardós. **ORB-SLAM2: an Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras**. *IEEE Transactions on Robotics,* vol. 33, no. 5, pp. 1255-1262, 2017.

ORBSLAM2



- На стереопаре мы сопоставляем точки между ракурсами, триангулируем, и сразу получаем 3Д координаты
- Выделяем «близкие» и «дальние» точки.
- Для близких достаточный параллакс позволяет точно оценить глубину и использовать для оценки позы камеры
- Дальние точки оцениваем только примерно, можем использовать для оценки поворотов

Raúl Mur-Artal and Juan D. Tardós. **ORB-SLAM2: an Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras**. *IEEE Transactions on Robotics,* vol. 33, no. 5, pp. 1255-1262, 2017.

Влияние стерео

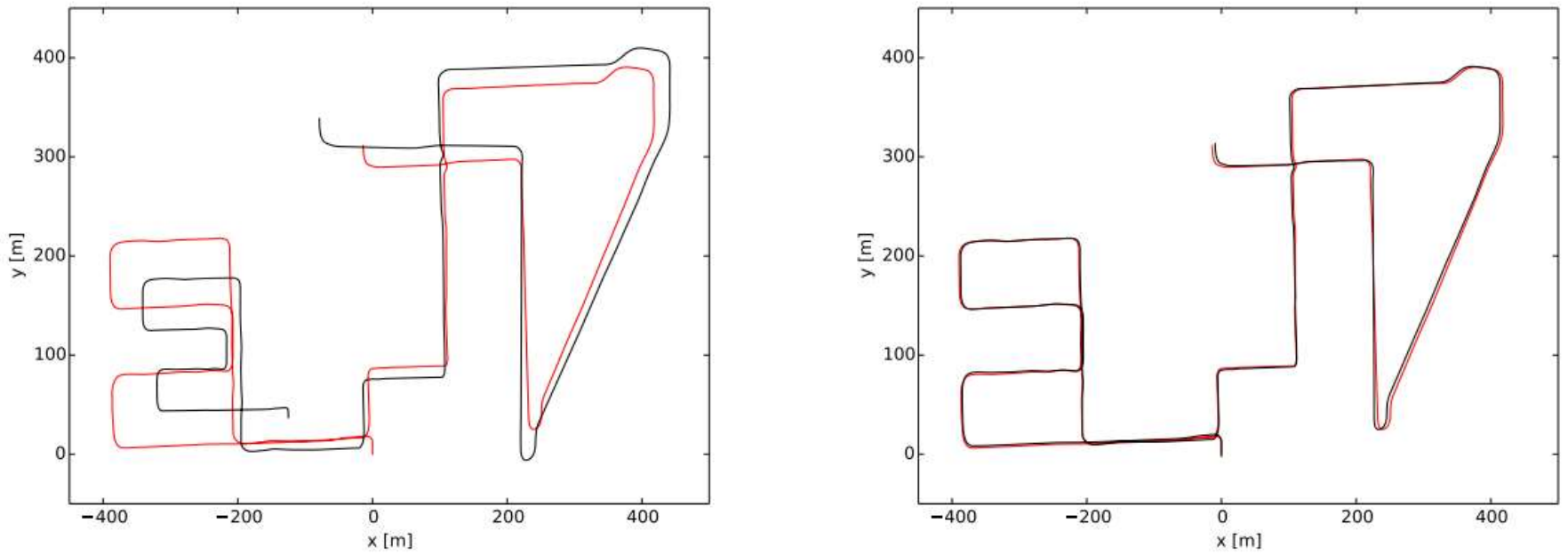
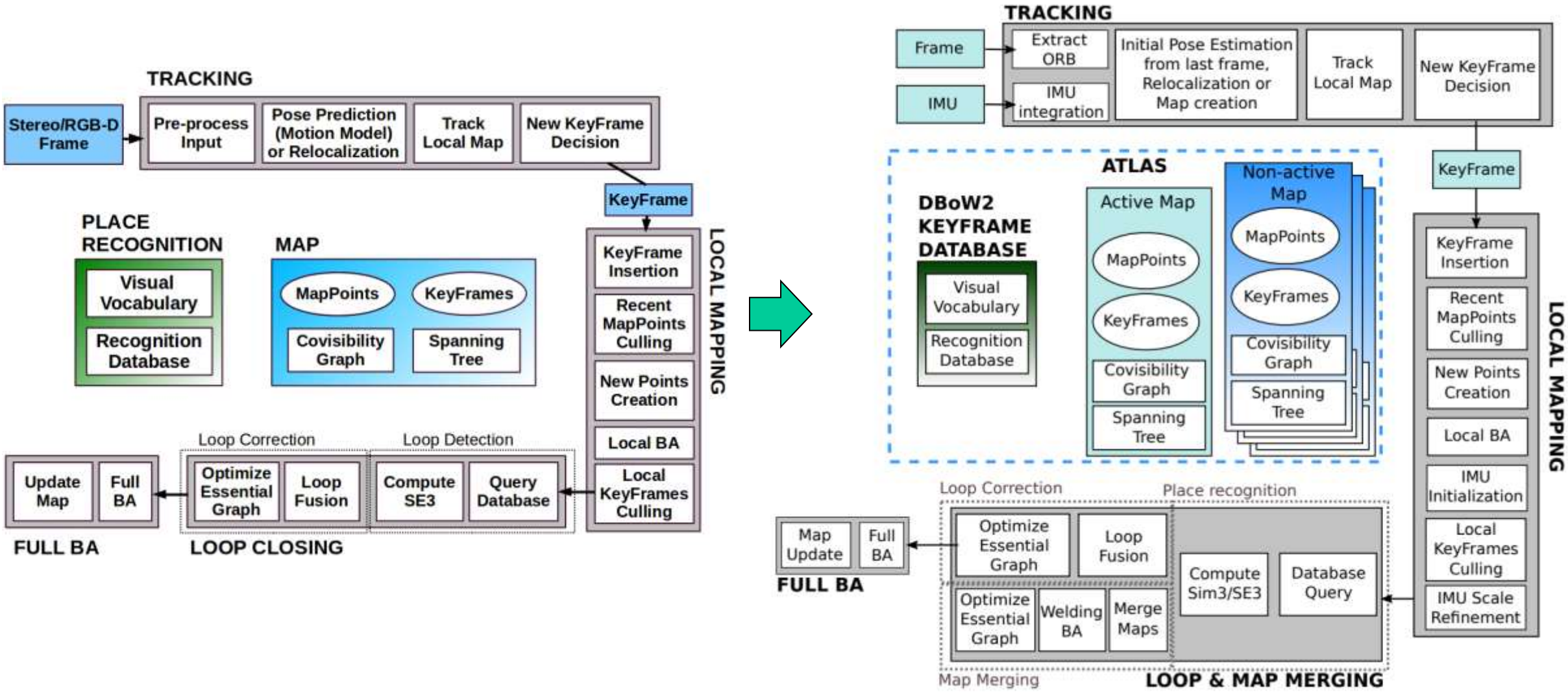


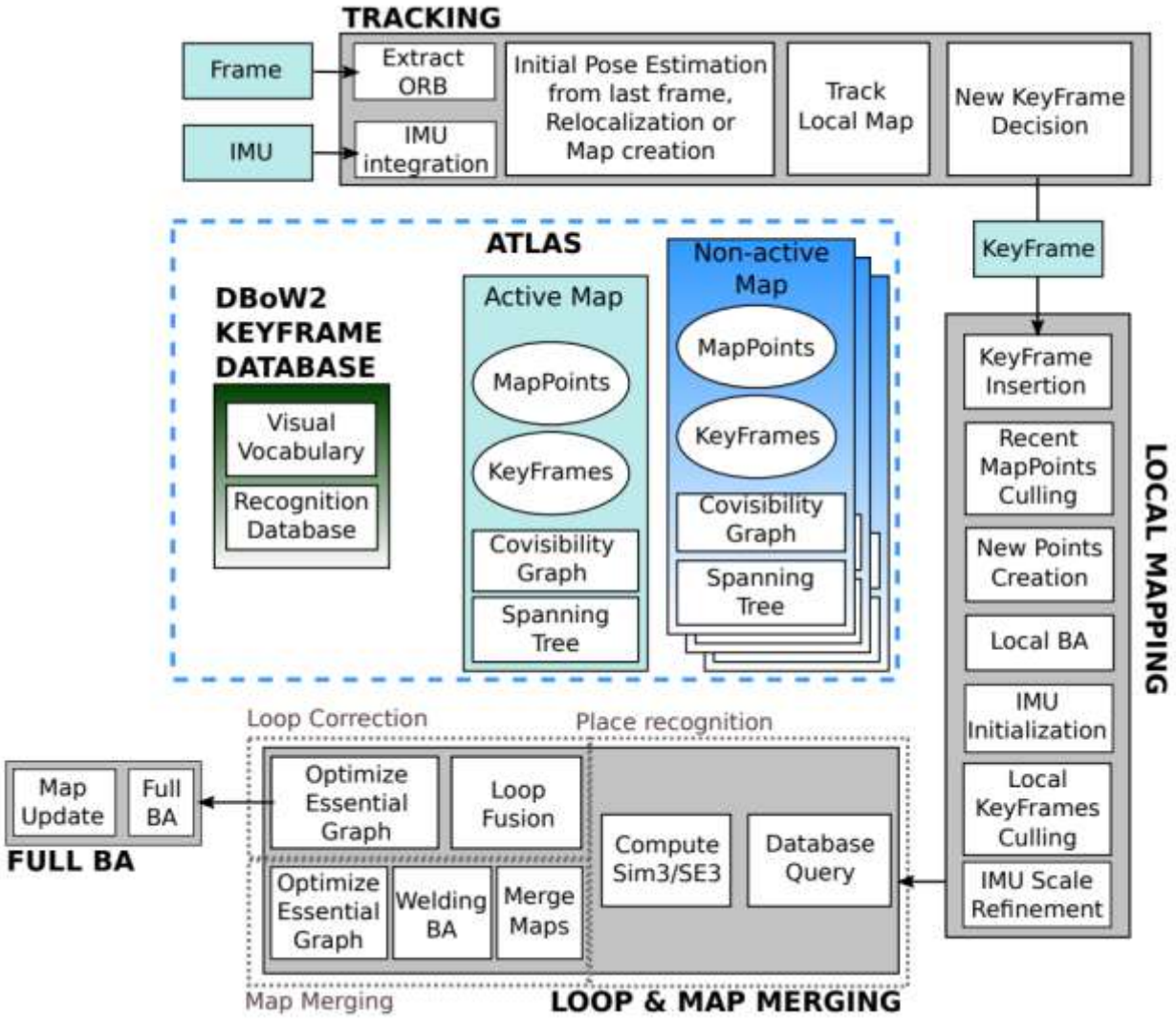
Fig. 5. Estimated trajectory (black) and ground-truth (red) in KITTI 08. Left: monocular ORB-SLAM [1], right: ORB-SLAM2 (stereo). Monocular ORB-SLAM suffers from severe scale drift in this sequence, especially at the turns. In contrast the proposed stereo version is able to estimate the true scale of the trajectory and map without scale drift.

ORBSLAM3



https://github.com/UZ-SLAMLab/ORB_SLAM3

ORBSLAM3



- Акселерометры (IMU) есть во многих устройствах (телефонах, камерах)
- IMU позволяет улучшить трекинг камеры при motion blur, малом числе ORB, и в целом повысить точность
- Atlas – карта, состоящая из многих отдельных карт. Одна из них «активна», используется для текущего трекинга.
- Если обнаруживаем «связь» между картами в атласе, тогда мы их объединяем

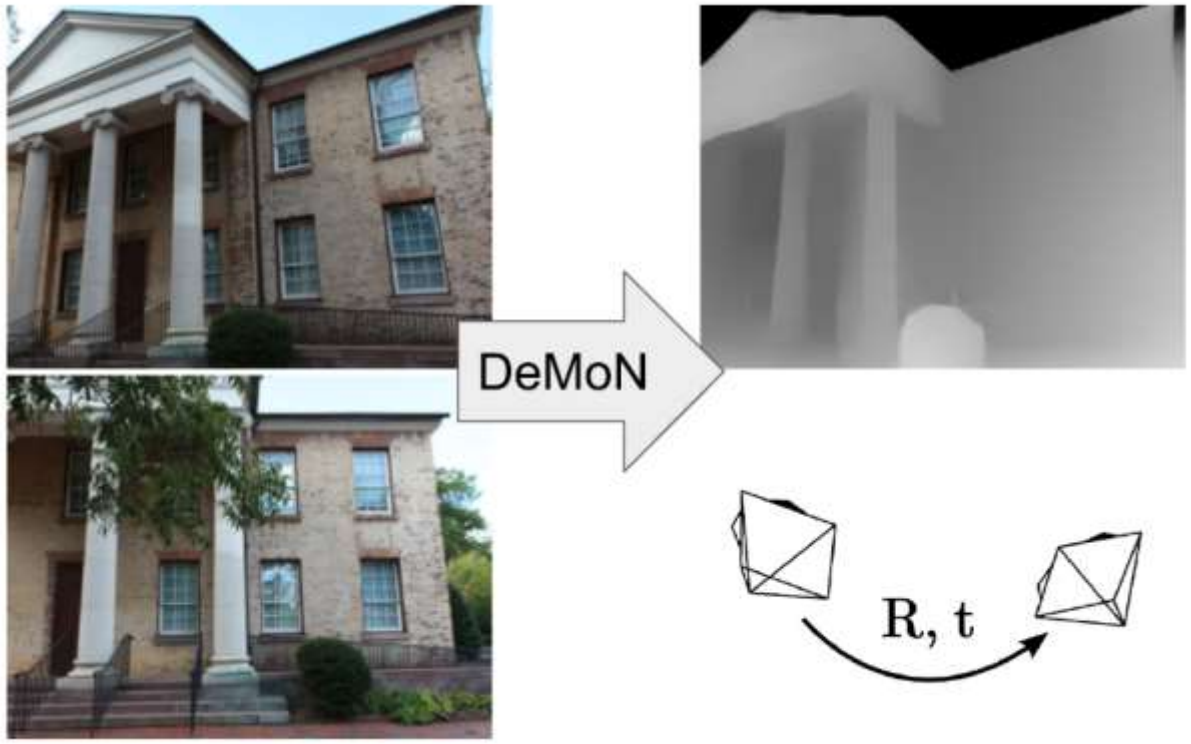
ORBSLAM3

Table II: Performance comparison in the EuRoC dataset (RMS ATE in m., scale error in %). Except where noted, we show results reported by the authors of each system, for all the frames in the trajectory, comparing with the processed GT.

			MH01	MH02	MH03	MH04	MH05	V101	V102	V103	V201	V202	V203	Avg[1]
Monocular	ORB-SLAM [4]	ATE[2,3]	0.071	0.067	0.071	0.082	**0.060**	**0.015**	0.020	-	**0.021**	**0.018**	-	0.047*
	DSO [27]	ATE	0.046	0.046	0.172	3.810	0.110	0.089	0.107	0.903	0.044	0.132	1.152	0.601
	SVO [24]	ATE	0.100	0.120	0.410	0.430	0.300	0.070	0.210	-	0.110	0.110	1.080	0.294*
	DSM [31]	ATE	0.039	0.036	0.055	**0.057**	0.067	0.095	0.059	0.076	0.056	0.057	**0.784**	**0.126**
	ORB-SLAM3 (ours)	ATE	**0.016**	**0.027**	**0.028**	0.138	0.072	0.033	**0.015**	**0.033**	0.023	0.029	-	0.041*
Stereo	ORB-SLAM2 [3]	ATE	0.035	**0.018**	0.028	0.119	0.060	**0.035**	**0.020**	**0.048**	**0.037**	0.035	-	0.044*
	VINS-Fusion [44]	ATE	0.540	0.460	0.330	0.780	0.500	0.550	0.230	-	0.230	0.200	-	0.424*
	SVO [24]	ATE	0.040	0.070	0.270	0.170	0.120	0.040	0.040	0.070	0.050	0.090	0.790	0.159
	ORB-SLAM3 (ours)	ATE	**0.029**	0.019	**0.024**	**0.085**	**0.052**	**0.035**	0.025	0.061	0.041	**0.028**	**0.521**	**0.084**
Monocular Inertial	MCSKF [33]	ATE[5]	0.420	0.450	0.230	0.370	0.480	0.340	0.200	0.670	0.100	0.160	1.130	0.414
	OKVIS [39]	ATE[5]	0.160	0.220	0.240	0.340	0.470	0.090	0.200	0.240	0.130	0.160	0.290	0.231
	ROVIO [42]	ATE[5]	0.210	0.250	0.250	0.490	0.520	0.100	0.100	0.140	0.120	0.140	0.140	0.224
	ORBSLAM-VI [4]	ATE[2,3] scale error[2,3]	0.075 0.5	0.084 0.8	0.087 1.5	0.217 3.5	0.082 0.5	**0.027** 0.9	0.028 0.8	- -	**0.032** 0.2	0.041 1.4	0.074 0.7	0.075* 1.1*
	VINS-Mono [7]	ATE[4]	0.084	0.105	0.074	0.122	0.147	0.047	0.066	0.180	0.056	0.090	0.244	0.110
	VI-DSO [46]	ATE scale error	**0.062** 1.1	0.044 0.5	0.117 0.4	0.132 0.2	0.121 0.8	0.059 1.1	0.067 1.1	0.096 0.8	0.040 1.2	0.062 0.3	0.174 0.4	0.089 0.7
	ORB-SLAM3 (ours)	ATE scale error	**0.062** 1.4	**0.037** 0.3	**0.046** 0.8	**0.075** 0.5	**0.057** 0.3	0.049 2.0	**0.015** 0.6	**0.037** 2.2	0.042 0.7	**0.021** 0.4	**0.027** 1.0	**0.043** 0.9
Stereo Inertial	VINS-Fusion [44]	ATE[4]	0.166	0.152	0.125	0.280	0.284	0.076	0.069	0.114	0.066	0.091	0.096	0.138
	BASALT [47]	ATE[3]	0.080	0.060	0.050	0.100	**0.080**	0.040	0.020	0.030	**0.030**	0.020	-	0.051*
	Kimera [8]	ATE	0.080	0.090	0.110	0.150	0.240	0.050	0.110	0.120	0.070	0.100	0.190	0.119
	ORB-SLAM3 (ours)	ATE scale error	**0.036** 0.6	**0.033** 0.2	**0.035** 0.6	**0.051** 0.2	0.082 0.9	**0.038** 0.8	**0.014** 0.6	**0.024** 0.8	0.032 1.1	**0.014** 0.2	**0.024** 0.2	**0.035** 0.6

Нейросетевой SLAM

Deep Visual Odometry

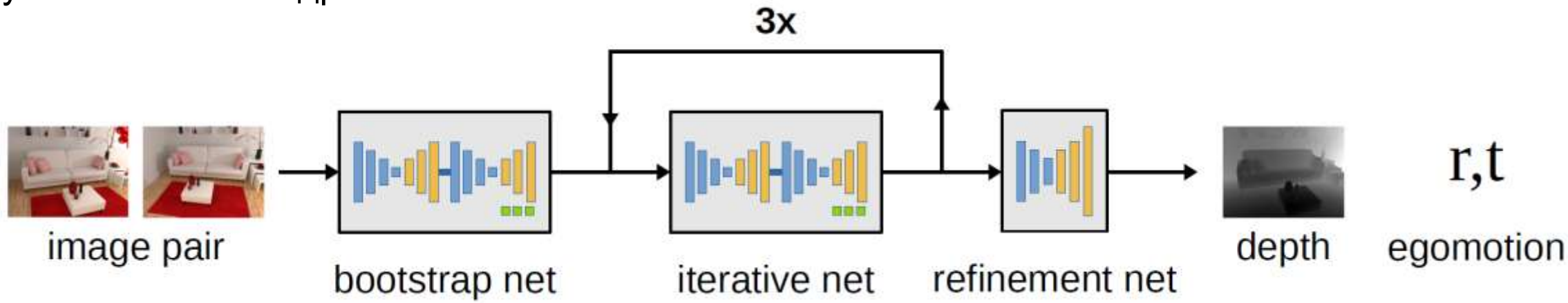


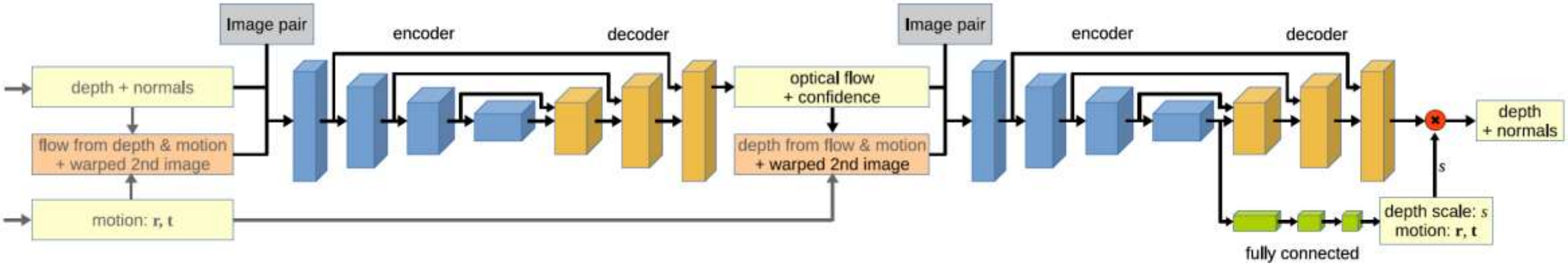
- Нейросетевые методы начали проникать в задачу SLAM постепенно, рассматривая отдельные подзадачи
- Пример - Visual Odometry
- Задача – оценить движение камеры между парой кадров
- Вспомогательные задачи, такие как оценка оптического потока, оценка карты глубины и т.д. используются как вспомогательные, чтобы нейросеть получала достаточно информации для обучения

Ummenhofer et. al. DeMoN: Depth and Motion Network for Learning Monocular Stereo. ICCV2017

DeMoN: Depth & Motion Network

Идея: чередуем оценку оптического потока и карты глубины для того, чтобы заставить сеть учитывать оба кадра

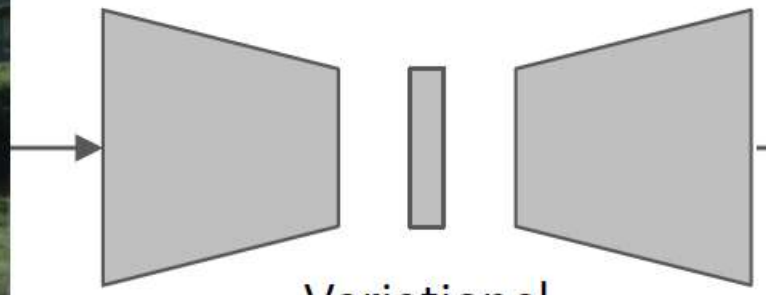




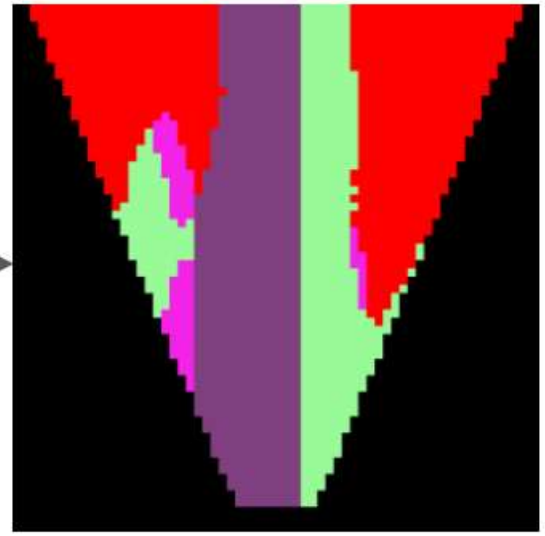
Ummenhofer et. al. DeMoN: Depth and Motion Network for Learning Monocular Stereo. ICCV2017

Deep mapping

Генерация семантической карты напрямую из 2D изображений



Variational
Encoder-Decoder Network



Input:
Monocular
RGB Image

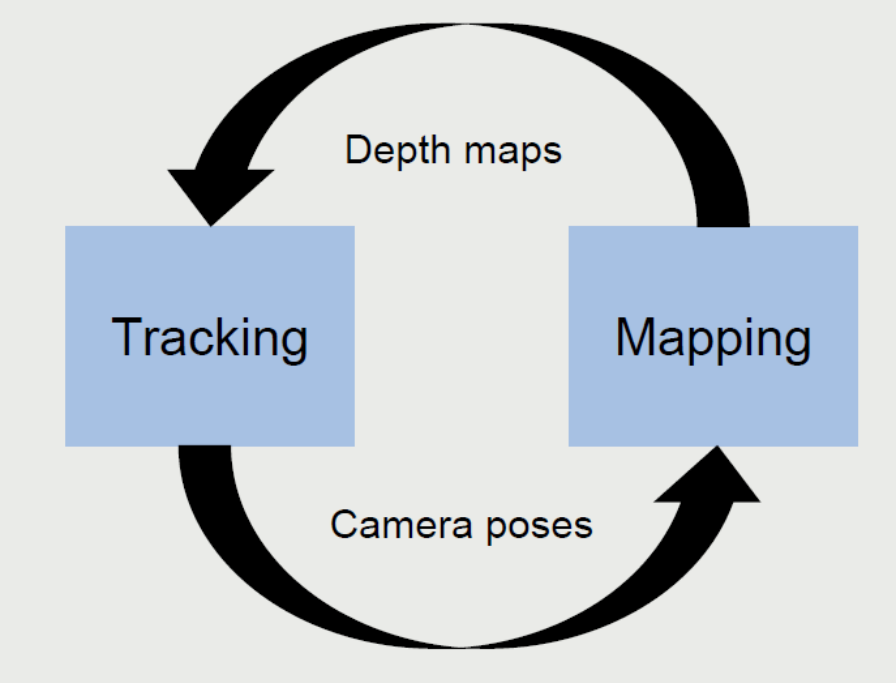
Legend:	Out of FOV
Road	Sidewalk
Terrain	Non free-space

Output:
Semantic-metric
Occupancy Grid Map

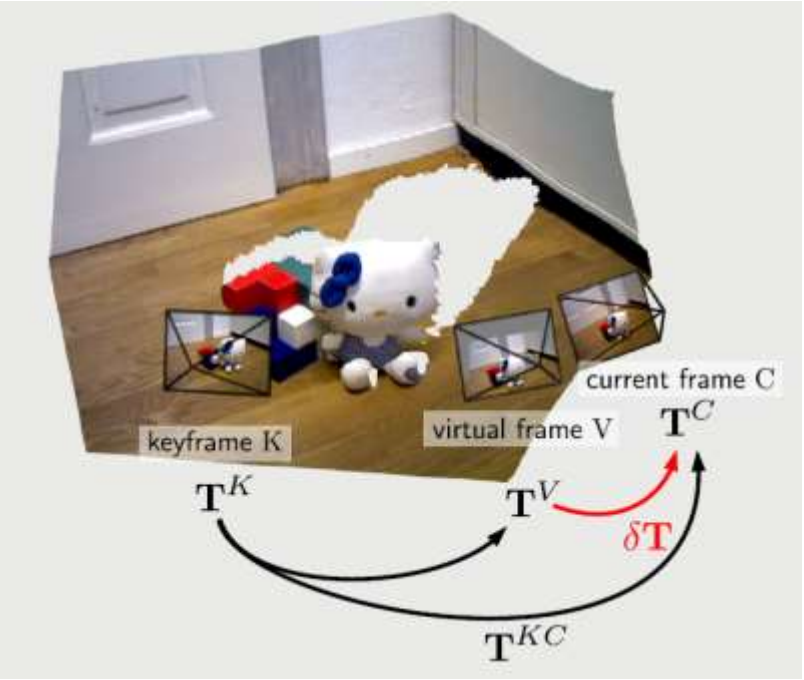
Source: arXiv:1804.02176v2

Deep Tracking and Mapping (DeepTAM)

Оценка позы на основе ключевых кадров и построение карты. Без loop closure и relocalization



Схема

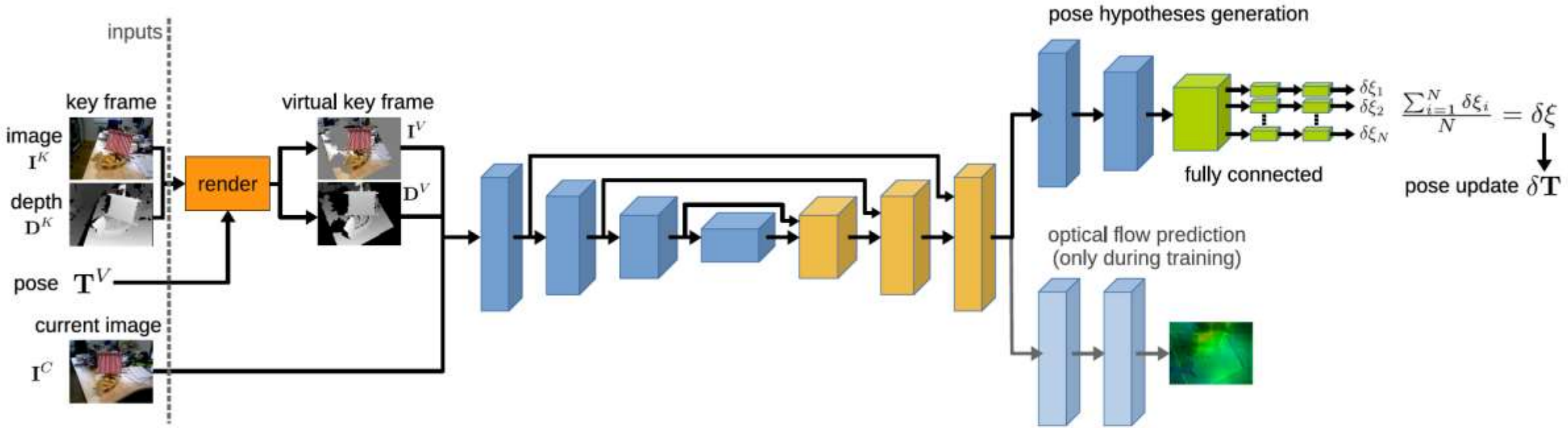


Оцениваем позу камеры относительно предсказанного виртуального кадра

H. Zhou et al., DeepTAM: Deep Tracking and Mapping. ECCV 2018

Модуль трекинга

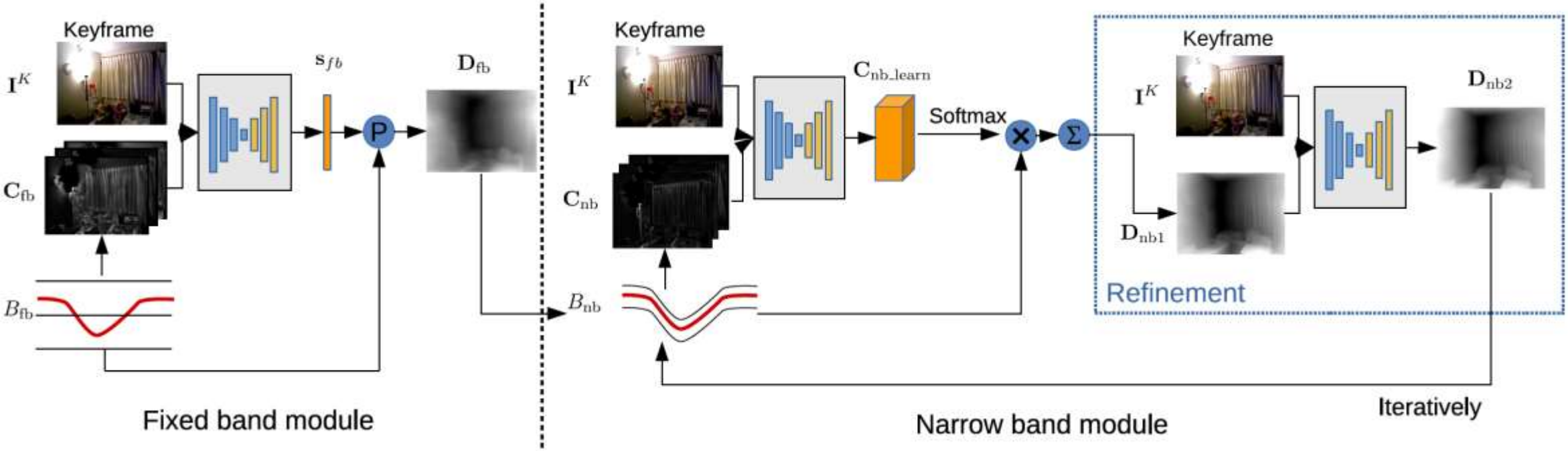
- Оцениваем позу и оптический поток
- Генерируем набор гипотез и затем усредняем их



H. Zhou et al., DeepTAM: Deep Tracking and Mapping. ECCV 2018

Модуль картографирования

- Mapping = high-quality depth estimation for keyframe



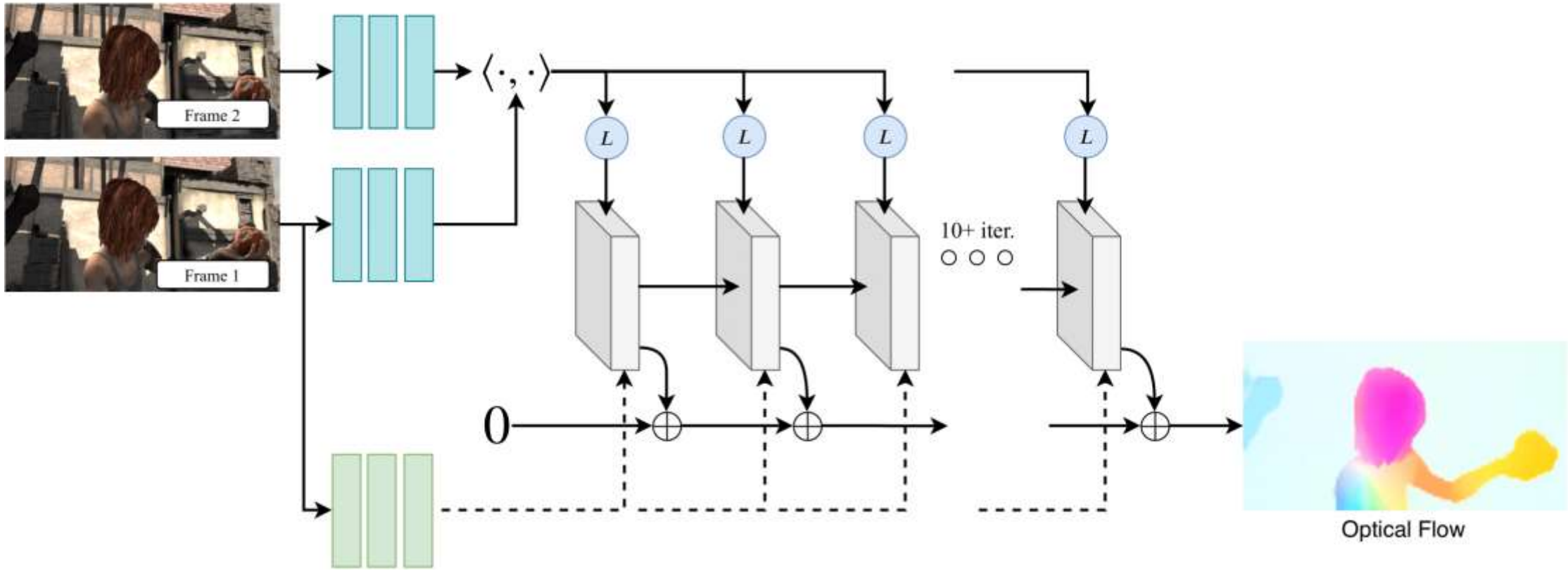
H. Zhou et al., DeepTAM: Deep Tracking and Mapping. ECCV 2018

Пример работы

H. Zhou et al., DeepTAM: Deep Tracking and Mapping. ECCV 2018

Recurrent All-Pairs Field Transforms (RAFT)

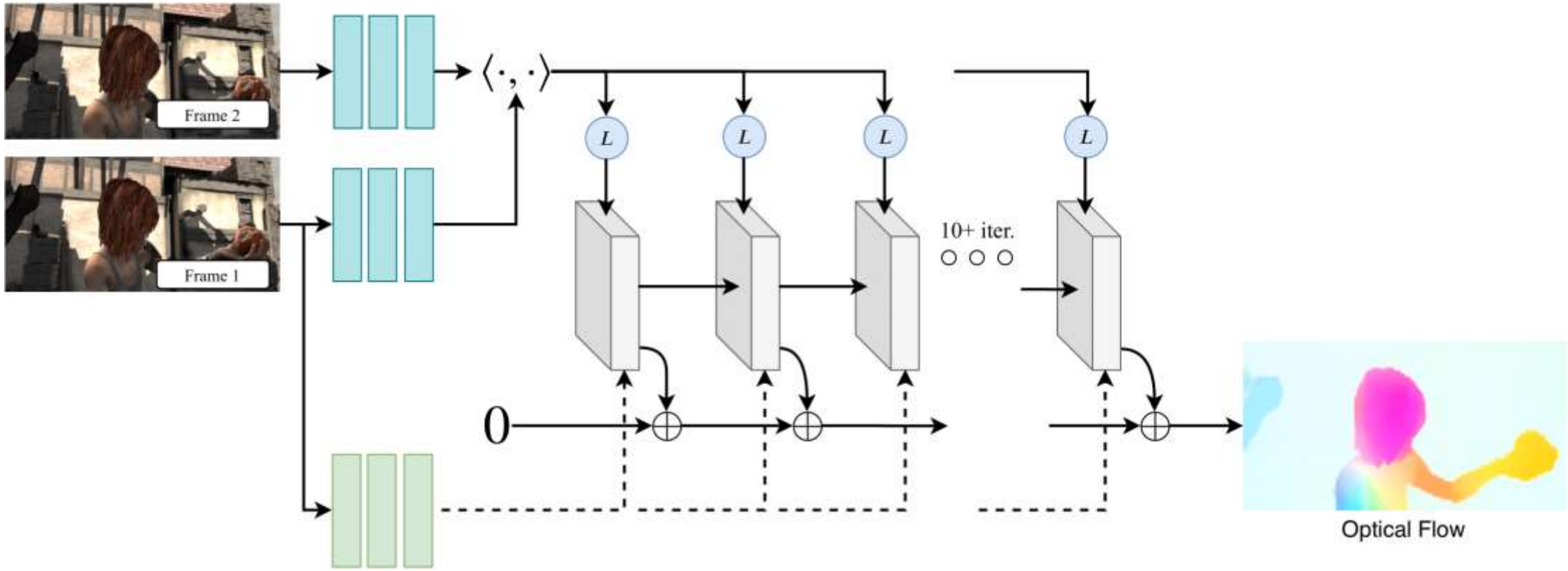
Метод оценки оптического потока на основе рекуррентного модуля



Zachary Teed, Jia Deng RAFT: Recurrent All-Pairs Field Transforms for Optical Flow. ECCV 2020

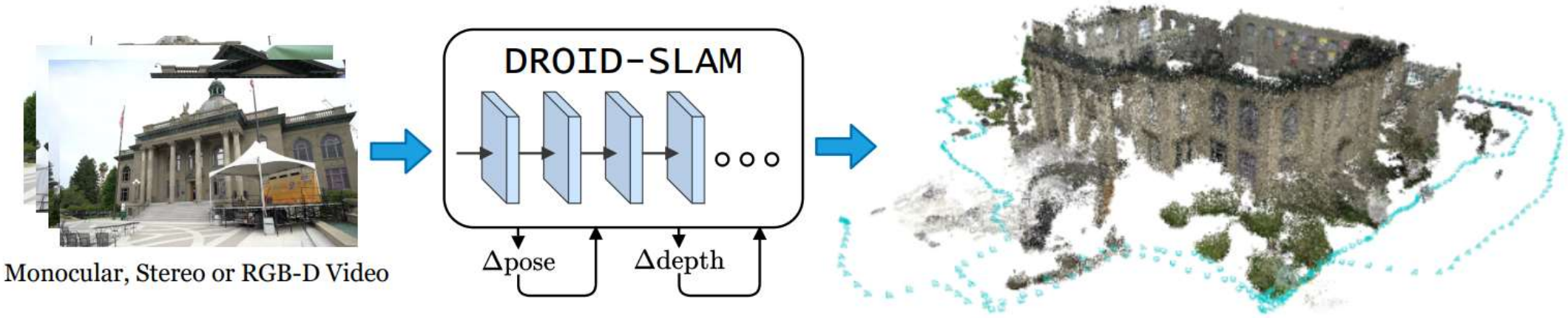
Recurrent All-Pairs Field Transforms (RAFT)

Метод оценки оптического потока на основе рекуррентного модуля



Zachary Teed, Jia Deng RAFT: Recurrent All-Pairs Field Transforms for Optical Flow. ECCV 2020

DROID-SLAM



- Полная схема SLAM, вдохновлённая идеями RAFT
- Будем строить граф сцены из набора ключевых кадров, для каждого оценивая позу и карту глубины.
- Нейросетевой модуль применяется к рёбрам графа, уточняя позу и карту глубины кадра

https://github.com/princeton-vl/DROID-SLAM

DROID-SLAM

- Система состоит из двух потоков – Frontend и Backend
- Frontend работает с потоком кадров.
  - Для нового кадра считают признаки, связывают с 3мя ближайшими ключевыми кадрами по оптическому потоку
  - Поза инициализируется линейной моделью
  - Оператор уточнения применяется несколько раз для уточнения позы и глубины нового ключевого кадра
- Backend – глобальный BA на всём графе сцены
  - Перестраиваем граф сцены, посчитав оптический поток между всеми парами кадров, инициализировав матрицу расстояний N*N
  - Проходим по всем рёбрам, начиная от близких по времени, и затем выбирая новые кадры в порядке увеличения оптического потока, подавляя близкие рёбра
  - Применяем оператор уточнения для всех ребёр получившегося графа

https://github.com/princeton-vl/DROID-SLAM

Резюме SLAM

- Задача SLAM близка к задаче SFM, но работает в итеративном он-лайн режиме, в реальном времени
- На практике пока в основном используются мультимодальные методы, работающие по классическому подходу с ключевыми точками
- Нейросетевые методы активно развиваются, превосходят по точности классические методы для случая RGB/RGBD камер, но пока очень вычислительно затратны