



ОБЗОР СОВРЕМЕННЫХ МЕТОДОВ ВИЗУАЛЬНОЙ ОДОМЕТРИИ

Терехов М. А.¹, студент, ✉ st054464@student.spbu.ru

¹Санкт-Петербургский государственный университет, Университетский проспект, д. 28, 198504, Старый Петергоф, Санкт-Петербург, Россия

Аннотация

В этой статье описываются задачи систем визуальной одометрии и SLAM и их основные применения. Далее перечисляются основные подходы, использованные научным сообществом для создания таких систем в разное время. Мы также углубляемся в более современный метод, основанный на совместной оптимизации, и разбираем его вариации в зависимости от требований к решению. Наконец, мы рассматриваем современные направления исследований в области визуальной одометрии и кратко представляем свои наработки.

Ключевые слова: визуальная одометрия, SLAM, автономная навигация, ADAS, UAV.

Цитирование: Терехов М. А. Обзор современных методов визуальной одометрии // Компьютерные инструменты в образовании. 2019. № 3. С. 5–14. doi:10.32603/2071-2340-2019-3-5-14

1. ВВЕДЕНИЕ

В попытках заставить автомобили перемещаться автономно, инженеры устанавливают на них внушительный набор датчиков. Ультразвуковые сонары отвечают за определение близких расстояний — например, при парковке. Лидары считывают более значительные дистанции, по ним можно определить, далеко ли остальные участники дорожного движения. Радары нацелены вперёд на максимальную дальность, чтобы как можно раньше запустить систему экстренного торможения при опасности. Камеры традиционно занимаются сегментацией и аннотированием, то есть размечают пространство: здесь дорожное полотно, там пешеход, а рядом — дорожный знак.

Оказывается, для решения многих из вышеперечисленных задач можно обойтись одной лишь камерой. В самом деле, если предположить, что перемещение нам известно, то простая тригонометрия позволяет по изменению положения объекта на кадре определить расстояние до него (рис. 1).

Восстановление геометрии пространства по информации с камер сейчас разрослось в множество продвинутых техник. Например, если нет ограничений по времени, задача получения трёхмерных моделей по набору снимков носит название *Structure from Motion* (SfM). В этой области исследователи добились небывалых результатов — плотные модели восстанавливаются по фотографиям, загруженным из интернета, без какой-либо информации о камере [7, 9].

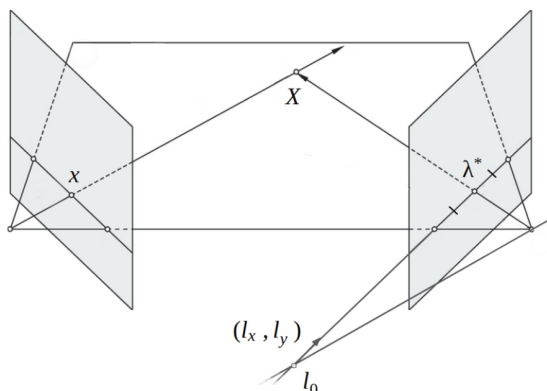


Рис. 1. Наблюдение точки X с двух различных положений камеры. С одной стороны, если знать положения точки x и λ^* на двух изображениях, можно получить её положение в пространстве. С другой, если знать положение точки x на первом изображении и глубину — расстояние от центра первой камеры до X , то можно выяснить λ^*

В свою очередь, системы *визуальной одометрии* (visual odometry, VO) занимаются восстановлением положения камеры и модели окружающего пространства, но по информации с видео. Обязательным является требование жёсткого реального времени, которое накладывает ограничения на возможную точность таких систем.

Из-за инкрементальной природы алгоритмов VO при их работе ошибка неизбежно накапливается. С этой проблемой можно бороться с помощью *замыкания циклов*. Если камера повторно посещает некоторое место, и при этом имеет возможность его «узнать», то траектория между посещениями может быть поправлена, замкнута в цикл. Алгоритмы VO с таким расширением традиционно носят название *визуальный SLAM* (Simultaneous Localization and Mapping, одновременная локализация и построение карты)¹.

Кроме описанных выше систем управления автомобилем, VO оказывается полезной в задачах автономной навигации беспилотных летательных аппаратов. Лидары и радары чрезмерно дороги и громоздки для установки на маленький дешёвый квадрокоптер, а ультразвуковые датчики измеряют расстояние только в одном направлении. Установка дешёвой камеры с VO на дрон позволяет определять расстояния до препятствий во всей области видимости и избегает проблем других датчиков.

2. Примеры систем визуальной одометрии

Первые системы VO основывались на принципах *фильтра Калмана* [13]. При этом набор точек на изображении рассматривался как зашумлённые измерения, а положения точек — как неизвестные². В дальнейшем такой подход был вытеснен системами на принципах *совместной оптимизации* (Bundle Adjustment, BA), так как последние показали более высокую точность [18].

Однако, в то время и в том, и в другом случае для работы алгоритмов требовались два связанных между собой компонента. Первый, *детектор*, должен выделять *ключе-*

¹Вообще, для задач SLAM успешно используются не только камеры. В случае наличия лидара решение уже доведено до промышленных стандартов. См, например, систему Google Cartographer [1].

²Подробнее про фильтр Калмана можно почитать в [11].



Рис. 2. Пример результата работы алгоритма DTAM [16] — плотного подхода к VO

вые точки на кадре так, чтобы при съёмках с другого ракурса выделились те же точки в пространстве. Второй, *дескриптор*, должен был соотносить ключевые точки на разных кадрах, соответствующие одному объекту. Эффективные детекторы и дескрипторы ключевых точек [17], хотя и могут работать с видео в реальном времени, не позволяют выделять более нескольких сотен точек на кадр. В приложениях, где требуется бóльшая плотность облака (например, при определении препятствий), ключевых точек не хватает.

Радикальным решением проблемы явились *плотные* методы. Они позволяют восстанавливать непрерывные поверхности вместо облаков точек (рис. 2), но требуют значительных вычислительных мощностей. Для работы в реальном времени подобной системе, скорее всего, потребуется GPU. Видеокарта с питанием не может быть установлена на дрон, поэтому возникает необходимость в промежуточном решении.

Подходы к VO, основанные на совместной оптимизации с использованием ключевых точек, называются *непрямыми*. Дело в том, что информация об изображении заменяется на этапе препроцессинга на информацию о ключевых точках, и дальнейшая работа ведётся только с ними. Известные примеры таких систем описаны в статьях [10, 14]. Альтернативой являются *прямые* методы, которые работают непосредственно с интенсивностями пикселей на кадрах. Идеи совместной оптимизации в прямых методах сохраняются, но внутренняя механика их работы существенно отличается от таковой в не прямых. Различия более подробно рассматриваются далее в тексте. За счёт того, что соотношение выбранных точек между кадрами не требуется, плотность облака в прямых методах ограничена только вычислительными ресурсами, а не наличием чётко различимых ключевых точек на кадре (рис. 3). Это позволяет определять препятствия с гораздо более высокой степенью надёжности. Известными примерами прямой VO могут служить работы [4, 5].

3. ПРЕДВАРИТЕЛЬНЫЕ СВЕДЕНИЯ

3.1. Представление движений

Как вам скажет любой математик, *движение* — функция $f : \mathbb{R}^3 \rightarrow \mathbb{R}^3$, сохраняющая расстояния. Как соотносить это определение с интуитивным пониманием движений? За-

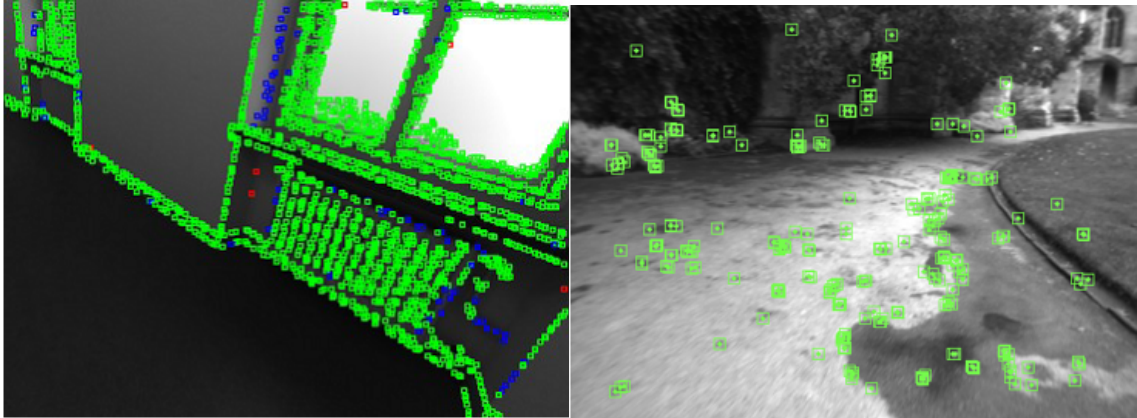


Рис. 3. Точки, отслеживаемые системой с прямым подходом (слева, алгоритм DSO [4]) в сравнении с точками, отслеживаемыми системой с непрямым (алгоритм ORB-SLAM [14])

фиксируем в пространстве *мировую систему координат* w , а также системы c_i , ассоциированные с i -м положением камеры. Теперь, чтобы перенести точку из мировой системы координат в систему i -го положения камеры, нам потребуется применить функцию $m_{w \rightarrow c_i}$:

$$[p]_{c_i} = m_{w \rightarrow c_i}([p]_w), \quad (1)$$

где $[p]_{coord}$ означает вектор координат точки p в системе координат $coord$.

Из теоремы Шаля известно, что любое движение, сохраняющее ориентацию, представимо как композиция поворота и параллельного переноса. Поворот можно представить умножением на ортогональную матрицу R , перенос — прибавлением вектора t . Всё вместе даёт

$$m_{a \rightarrow b}(v) = R_{a \rightarrow b}v + t_{a \rightarrow b}. \quad (2)$$

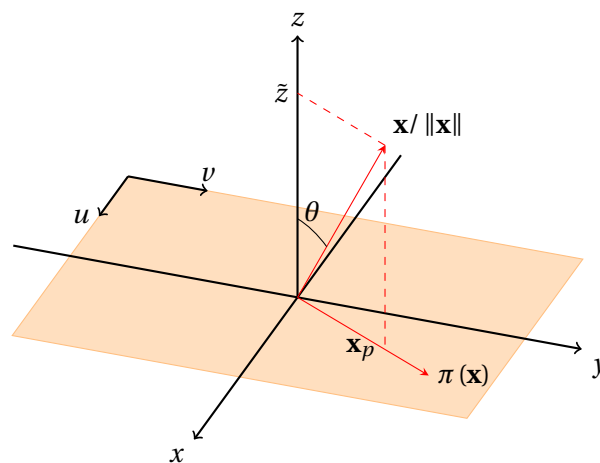


Рис. 4. Проецирование точки x на изображение

3.2. Модель камеры

Для того, чтобы соотнести точку в пространстве и координаты пикселя на изображении, требуется функция $\pi : \mathbb{R}^3 \rightarrow \Omega$, называемая *моделью камеры*. Здесь $\Omega \subset \mathbb{R}^2$ — область изображения. Геометрическим местом точек пространства, отображаемых камерой в одну точку на изображении, является луч. В большинстве случаев достаточно так называемых *центрально-проективных* моделей, в которых все такие лучи имеют общее начало, называемое центром камеры. В нём располагается система координат, связанная с камерой (рис. 4). Обратное, для получения точки в пространстве по точке на изображении, используется функция $\pi^{-1} : \Omega \times \mathbb{R}_+ \rightarrow \mathbb{R}^3$. Ей требуется *глубина* точки, то есть расстояние от неё до центра камеры. По понятным причинам, без глубины не получится восстановить положение точки в пространстве.

О том, какие бывают модели камеры, можно узнать из замечательной книги [8].

4. СОВМЕСТНАЯ ОПТИМИЗАЦИЯ

4.1. Непрямой подход

Ключевая идея совместной оптимизации заключается в «развороте» задачи одометрии. Если предположить, что мы приблизительно знаем траекторию камеры и положения точек, которые она наблюдает, в пространстве, то мы можем оценить «хорошесть» нашего предположения. Действительно, коль скоро мы знаем движение между двумя кадрами и глубины точек на одном из них, мы можем предсказать, где эта точка окажется на втором кадре. С другой стороны, из соответствия ключевых точек уже знаем, где она должна оказаться. Отсюда в простейшем случае получается следующая *ошибка репроекции*:

$$\text{err}_g(m_{c_1 \rightarrow c_2}, d_1, \dots, d_n) = \sum_{p_i \in P} \|\pi(m_{c_1 \rightarrow c_2}(\pi^{-1}(p_i, d_i))) - q_i\|^2, \quad (3)$$

где $m_{c_1 \rightarrow c_2}$ — движение от первого положения камеры ко второму, $p_i \in P$ — точки на первом кадре, d_i — соответствующие им глубины, q_i — найденные соответствия на втором кадре.

Теперь, чтобы решить исходную задачу, то есть определить движение $m_{c_1 \rightarrow c_2}^*$ и глубины d_i^* (что эквивалентно построению облака), нужно подобрать их минимизирующими функцию err_g .

4.2. Прямой подход

Описанная выше функция err_g использует известные заранее соответствия q_i на втором кадре. Чтобы отказаться от их использования, нам нужна принципиально иная модель ошибок. А именно, мы будем накладывать на модель требование *фотометрической консистентности*, то есть будем ожидать, что если мы наблюдаем одну и ту же точку в пространстве с разных ракурсов, *интенсивность* соответствующих пикселей на разных кадрах будет совпадать. Вооружившись этой идеей, можем записать следующую функцию ошибок:

$$\text{err}_{ph}(m_{c_1 \rightarrow c_2}, d_1, \dots, d_n) = \sum_{p_i \in P} |I_2[\pi(m_{c_1 \rightarrow c_2}(\pi^{-1}(p_i, d_i)))] - I_1[p_i]|^2, \quad (4)$$

где $I_1, I_2 : \Omega \rightarrow \mathbb{R}$ — функции, соответствующие первому и второму кадрам. Чтобы из таблицы значений, которой представляется изображение, получить непрерывную на всём Ω функцию, используется *интерполяция*.

4.3. Детали реализации

Алгоритмы оптимизации

Задача минимизации подобных *целевых функций* (как в прямом, так и в непрямом случае) носит название *нелинейной оптимизации*, и существует множество методов её решения. В данном случае традиционно применяются алгоритм Гаусса-Ньютона или более продвинутый алгоритм Левенберга-Марквардта. Оба метода непосредственно опираются на дифференцирование целевой функции, чтобы аппроксимировать её значение в окрестности текущего решения и понять, как его можно улучшить.

Программная реализация

Первое успешное применение идей совместной оптимизации к задаче VO принадлежит работе PTAM (Parallel Tracking and Mapping) [10], авторы которой предложили несколько важных идей. Во-первых, они выделили из видео разреженное множество *ключевых кадров*, и стали определять глубины только для точек на кадрах из этого множества. Частота добавления ключевых кадров позволяет регулировать количество вычислений для совместной оптимизации, чтобы она не отставала от реального времени. Во-вторых, они разделили выполнение на два потока, о чём свидетельствует название системы. В первом потоке производится отслеживание (*tracking*) новых кадров относительно уже построенного облака точек. Этот поток жёстко привязан к реальному времени, но быстрый трекинг обеспечить сравнительно просто. Во втором потоке производится обновление облака точек (*mapping*), связанное с более громоздкой задачей оптимизации, и этот поток может по временам отставать от текущего видео, с надеждой догнать позднее. Сейчас такое разделение используется повсеместно.

5. ТЕКУЩИЕ НАПРАВЛЕНИЯ РАЗВИТИЯ И НАША РАБОТА

Задачи визуальной одометрии и SLAM — сравнительно новые для научного сообщества, поскольку не так давно вычислительные мощности доросли до уровня, когда подобные задачи могли бы быть решены в реальном времени. Из-за этого соответствующая научная область ещё не так обширно исследована, есть место для новой работы.

Читатель не сильно удивится, если узнает, что одним из популярных направлений исследований в области является применение машинного обучения. Конкретнее, высокие результаты показывает предсказывание глубин с помощью нейронных сетей с различной архитектурой: GAN в [3], CNN в [19, 23].

Другим направлением развития является расширение набора конфигураций оборудования, на котором должна работать одометрия. Многие алгоритмы получилось существенно улучшить, если использовать видео со стерео-пары камер [6, 15, 22]. Также существующие алгоритмы расширяются для работы с широкоугольными камерами, поскольку именно такие встречаются на практике [12].

Но в случае с описанным в начале примером с автомобилем требуется более общая конфигурация. В рамках технологии surround view на автомобиле устанавливается система из четырёх камер, направленных в разные стороны (рис. 5). Обычно такие камеры используются, например, чтобы генерировать вид сверху на автомобиль для удобства водителя. Подача видео с этих камер на вход алгоритму визуальной одометрии позволила бы расширить их область применения. В таком случае камеры могли бы использоваться для задач, традиционно ассоциированных с другими датчиками: экстренное торможение, определение препятствий, помощь при парковке и т. д.

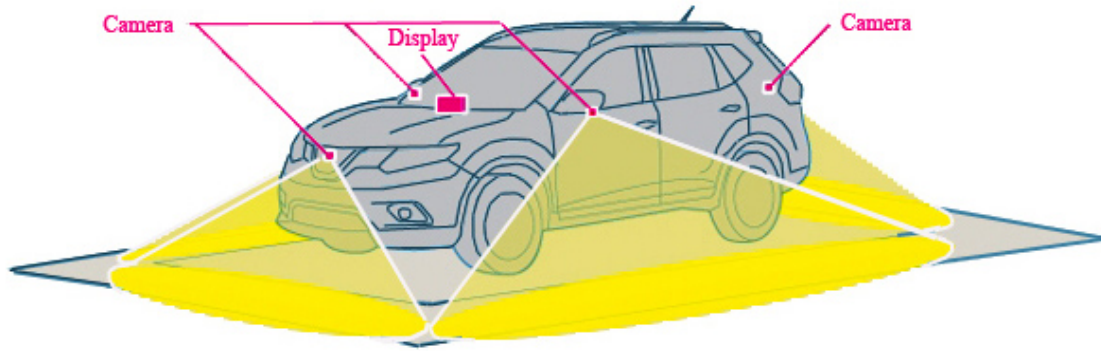


Рис. 5. Положение камер в типичной системе surround view

Ещё одним интересным применением данного метода может служить продолжение траектории. В случае, если автомобиль въезжает в подземный туннель, сигнал GPS пропадает. Применение технологии одометрии позволит на протяжении нескольких километров аккуратно отслеживать положение автомобиля и показывать траекторию водителю. Аналогично можно рассмотреть пример подземного паркинга.

Далее, хорошим примером является автономная навигация дронов. Чтобы точно восстанавливать траекторию дрона с помощью камер современными методами, требуется стерео-пара из камер, направленных в одну сторону. Такая конфигурация «видит» лишь в одном направлении. Но если бы камеры смотрели в разных направлениях, обзор мог бы составлять до 360 градусов. Это позволило бы определять препятствия — высотные дома, холмы и т. п. — со всех сторон от дрона.

Для работы с такой конфигурацией требуется алгоритм, способный использовать как единое целое изображения с широкоугольных камер с потенциально непересекающимися углами обзора. На данный момент не так много систем были расширены для работы в таком случае [20, 21]. Более того, ни одна из приведённых работ не использует прямой подход, поэтому облака точек получаются разреженными. Таким образом, задача построения полуплотного облака точек и восстановления траектории автомобиля по видео с системы камер в реальном времени до сих пор не решена.

Мы пытаемся заполнить образовавшуюся нишу и реализуем расширение подхода DSO (Direct Sparse Odometry) [4] на случай системы широкоугольных камер. Расширение для одной такой камеры уже было описано [12], но авторы не предоставили исходного кода, поэтому нам пришлось частично повторять их результаты. Сейчас нами создан прототип системы, работающий с одной камерой, и мы разрабатываем алгоритм трекинга, который бы позволил быстро отслеживать положения системы камер относительно известного облака точек. Исходный код проекта доступен по адресу [2].

Список литературы

1. Google cartographer. URL: <https://github.com/googlecartographer/cartographer> (дата обращения: 07.09.2019).
2. Multicamera dso. URL: <https://github.com/MikhailTerekhov/mdso> (дата обращения: 07.09.2019).
3. *Almalioglu Y., Saputra M. R. U., de Gusmao P. P. B., Markham A., Trigoni N.* Ganvo: Unsupervised deep monocular visual odometry and depth estimation with generative adversarial networks // 2019 International Conference on Robotics and Automation (ICRA) (2019). IEEE. P. 5474–5480. doi: 10.1109/ICRA.2019.8793512
4. *Engel J., Koltun V., Cremers D.* Direct sparse odometry // IEEE transactions on pattern analysis and machine intelligence. 2018. Vol. 40. № 3. P. 611–625. doi: 10.1109/TPAMI.2017.2658577

5. Engel J., Schöps T., Cremers D. Lsd-slam: Large-scale direct monocular slam // Computer Vision — ECCV 2014. Zurich: Springer, 2014. P. 834–849. doi: 10.1007/978-3-319-10605-2_54
6. Engel J., Stückler J., Cremers D. Large-scale direct slam with stereo cameras // 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE. 2015. P. 1935–1942. doi: 10.1109/IROS.2015.7353631
7. Frahm J., et al. Building rome on a cloudless day // Computer Vision — ECCV 2010. Heraklion, Crete, Greece. 2010. P. 368–381. doi: 10.1109/IROS.2015.7353631
8. Hartley R., Zisserman A. Multiple view geometry in computer vision. Cambridge: Cambridge university press, 2003.
9. Heinly J., Schönberger J., Dunn E., Frahm J. Reconstructing the world* in six days *(as captured by the yahoo 100 million image dataset) // The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Boston: MA, 2015. P. 3287–3295.
10. Klein G., Murray D. Parallel tracking and mapping for small ar workspaces // Proc. of the 2007 6th IEEE and ACM Int. Symp. on Mixed and Augmented Reality. Washington: DC, 2007. P. 1–10. doi: 10.1109/ISMAR.2007.4538852
11. Lacey T. Tutorial: The kalman filter. URL: web.mit.edu/kirtley/kirtley/binlustuff/literature/control/Kalmanfilter.pdf (дата обращения: 07.09.2019).
12. Matsuki H., von Stumberg L., Usenko V., Stückler J., Cremers, D. Omnidirectional dso: Direct sparse odometry with fisheye cameras // IEEE Robotics and Automation Letters. 2018. Vol. 3. № 4. P. 3693–3700.
13. Mourikis A. I., Roumeliotis S. I. A multi-state constraint kalman filter for vision-aided inertial navigation // Proceedings 2007 IEEE International Conference on Robotics and Automation (2007). IEEE, 2007. P. 3565–3572. doi: 10.1109/ROBOT.2007.364024
14. Mur-Artal R., Montiel J., Tardós J. Orb-slam: A versatile and accurate monocular slam system // Transactions on Robotics. 2015). Vol. 31. № 5. P. 1147–1163. doi: 10.1109/TRO.2015.2463671
15. Mur-Artal R., Tardós J. D. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras // IEEE Transactions on Robotics. 2017. Vol. 33. № 5. P. 1255–1262. doi: 10.1109/TRO.2017.2705103
16. Newcombe R. A., Lovegrove S. J., Davison A. J. Dtam: Dense tracking and mapping in real-time // 2011 International Conference on Computer Vision (2011), Barcelona, 2011. P. 2320–2327. doi: 10.1109/ICCV.2011.6126513
17. Rublee E., Rabaud V., Konolige K., Bradski G. Orb: An efficient alternative to sift or surf // European Conference on Computer Vision. 2011. P. 2564–2571.
18. Strasdat H., Montiel J. M., Davison A. J. Visual slam: why filter? // Image and Vision Computing. 2012. Vol. 30. № 2. P. 65–77. doi: 10.1016/j.imavis.2012.02.009
19. Tateno K., Tombari F., Laina I., Navab N. Cnn-slam: Real-time dense monocular slam with learned depth prediction // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017). Hawaii, 2017. P. 6243–6252.
20. Tribou M. J., Harmat A., Wang D. W., Sharf I., Waslander S. L. Multi-camera parallel tracking and mapping with non-overlapping fields of view // The International Journal of Robotics Research, 2015. Vol. 34. № 12. P. 1480–1500. doi: 10.1177/0278364915571429
21. Urban S., Hinz S. Multicol-slam-a modular real-time multi-camera slam system // arXiv preprint arXiv:1610.07336. 2016. URL: <https://arxiv.org/abs/1610.07336> (дата обращения: 07.09.2019).
22. Wang R., Schworer M., Cremers D. Stereo dso: Large-scale direct sparse visual odometry with stereo cameras // Proceedings of the IEEE International Conference on Computer Vision. 2017. P. 3903–3911.
23. Yang N., Wang R., Stückler J., Cremers D. Deep virtual stereo odometry: Leveraging deep depth prediction for monocular direct sparse odometry // Proceedings of the European Conference on Computer Vision (ECCV). 2018. P. 817–833.

Поступила в редакцию 15.08.2019, окончательный вариант — 07.09.2019.

Терехов Михаил Андреевич, студент четвёртого курса математико-механического факультета СПбГУ, ✉ st054464@student.spbu.ru

Computer tools in education, 2019

№ 3: 5–14

<http://cte.eltech.ru>

doi:10.32603/2071-2340-2019-3-5-14

Overview of Modern Approaches to Visual Odometry

Terekhov M. A.¹, student, ✉ st054464@student.spbu.ru

¹Saint Petersburg State University, 28, Universitetsky pr., 198504, Stary Peterhof, Saint Petersburg, Russia

Abstract

In this paper we describe the tasks of Visual Odometry and Simultaneous Localization and Mapping systems along with their main applications. Next, we list some approaches used by the scientific community to create such systems in different time periods. We then proceed to explain in detail the more recent method based on bundle adjustment and show some of its variations for different applications. At last, we overview present-day research directions in the field of visual odometry and briefly present our work.

Keywords: *visual odometry, SLAM, autonomous navigation, ADAS, UAV.*

Citation: M. A. Terekhov, "Overview of Modern Approaches to Visual Odometry," *Computer tools in education*, no. 3, pp. 5–14, 2019 (in Russian); doi:10.32603/2071-2340-2019-3-5-14

References

1. *Google cartographer*, [Online], Available: <https://github.com/googlecartographer/cartographer>
2. "Multicamera dso," *GitHub*, [Online], Available: <https://github.com/MikhailTerekhov/mdso>
3. Y. Almalioglu, M. R. U. Saputra, P. P. B. de Gusmão, A. Markham, and N. Trigoni, "GANVO: Unsupervised Deep Monocular Visual Odometry and Depth Estimation with Generative Adversarial Networks," in *2019 International Conference on Robotics and Automation (ICRA)*, Montreal, QC, Canada, 2019, pp. 5474–5480; doi: 10.1109/ICRA.2019.8793512
4. J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 3, pp. 611–625, 2018; doi: 10.1109/TPAMI.2017.2658577
5. J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *Computer Vision – ECCV 2014*, Zurich, Switzerland, pp. 834–849, 2014; doi: 10.1007/978-3-319-10605-2_54
6. J. Engel, T. Schöps, and D. Cremers, "Large-scale direct SLAM with stereo cameras," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, 2015, pp. 1935–1942; doi: 10.1109/IROS.2015.7353631
7. J.-M. Frahm et al., "Building Rome on a cloudless day," in *Computer Vision – ECCV 2010*, Heraklion, Crete, Greece, 2010, pp. 368–381; doi: 10.1007/978-3-642-15561-1_27
8. R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*, Cambridge, UK: Cambridge university press, 2003.
9. J. Heinly, J. L. Schonberger, E. Dunn, and J.-M. Frahm, "Reconstructing the world* in six days *(as captured by the yahoo 100 million image dataset)," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, 2015, pp. 3287–3295.
10. G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *Proc. of the 2007 6th IEEE and ACM Int. Symp. on Mixed and Augmented Reality*, Washington, DC, 2007, pp. 1–10; doi: 10.1109/ISMAR.2007.4538852
11. T. Lacey, "Tutorial: The kalman filter," MIT, [Online], Available: <http://web.mit.edu/kirtley/kirtley/binlustuff/literature/control/Kalman%20filter.pdf>

12. H. Matsuki, L. von Stumberg, V. Usenko, J. Stückler, and D. Cremers, “Omnidirectional DSO: Direct Sparse Odometry With Fisheye Cameras,” *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3693–3700, 2018; doi: 10.1109/LRA.2018.2855443
13. A. I. Mourikis and S. I. Roumeliotis, “A multi-state constraint kalman filter for vision-aided inertial navigation,” in *Proc. 2007 IEEE Int. Conf. on Robotics and Automation*, Roma, Italy, 2007, pp. 3565–3572; doi: 10.1109/ROBOT.2007.364024
14. R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, “Orb-slam: A versatile and accurate monocular slam system,” *Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015; doi: 10.1109/TRO.2015.2463671
15. R. Mur-Artal and J. D. Tardós, “ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras,” *Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017; doi: 10.1109/TRO.2017.2705103
16. R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, “DTAM: Dense tracking and mapping in real-time,” in *2011 International Conference on Computer Vision*, Barcelona, Spain, 2011, pp. 2320–2327; doi: 10.1109/ICCV.2011.6126513
17. E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “Orb: An efficient alternative to sift or surf,” in *European Conf. on Computer Vision*, 2011, pp. 2564–2571.
18. H. Strasdat, J. M. M. Montiel, and A. J. Davison, “Visual slam: why filter?” *Image and Vision Computing*, vol. 30, no. 2, pp. 65–77, 2012; doi: 10.1016/j.imavis.2012.02.009
19. K. Tateno, F. Tombari, I. Laina, and N. Navab, “Cnn-slam: Real-time dense monocular slam with learned depth prediction,” in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Hawaii, USA, 2017, pp. 6243–6252.
20. M. J. Tribou, A. Harmat, D. W. L. Wang, I. Sharf, and S. L. Waslander, “Multi-camera parallel tracking and mapping with non-overlapping fields of view,” *The International Journal of Robotics Research*, vol. 34, no. 12, pp. 1480–1500, 2015; doi: 10.1177/0278364915571429
21. S. Urban and S. Hinz, “Multicol-slam-a modular real-time multi-camera slam system,” in arXiv preprint, arXiv:1610.07336, 2016, [Online], Available: <https://arxiv.org/abs/1610.07336>
22. R. Wang, M. Schworer, and D. Cremers, “Stereo dso: Large-scale direct sparse visual odometry with stereo cameras,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3903–3911.
23. N. Yang, R. Wang, J. Stückler, and D. Cremers, “Deep virtual stereo odometry: Leveraging deep depth prediction for monocular direct sparse odometry,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 817–833.

Received 15.08.2019, the final version — 07.09.2019.

Mikhail A. Terekhov, fourth year bachelor student, Mathematics and Mechanics faculty, Saint Petersburg State University, ✉ st054464@student.spbu.ru