

УДК 629.7.02n
ББК 39.52
Л91

Шаошань Лю, Лиюнь Ли, Цзе Тан, Шуаш Ву, Жан-Люк Годье
Л91 Разработка беспилотных транспортных средств / науч. ред. В. С. Яценков;
пер. с англ. П. М. Бомбаковой. – М.: ДМК Пресс, 2021. – 242 с.: ил.

ISBN 978-5-97060-969-9

В книге представлен один из первых технических обзоров беспилотных транспортных средств (БТС), адресованных широкой аудитории компьютерных специалистов и инженеров, а также студентов технических вузов и всех, кто сталкивается с БТС на практике.

Авторы делятся практическим опытом разработки БТС – начиная с алгоритмов локализации, восприятия, планирования и контроля и заканчивая глубоким обучением моделей. Рассматриваются последние достижения в области управления БТС, а также практические примеры из реальной жизни, что позволит читателям эффективно применять полученные знания в коммерческих проектах.

В разделе «Преподавание и обучение» множество полезной информации, которая поможет использовать книгу как часть программы учебных курсов.

УДК 629.7.02
ББК 39.52

© 2020 Morgan and Claypool Publishers All Rights Reserved Morgan and Claypool Publishers

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

ISBN (англ.) 978-1-68173-935-9
ISBN (рус.) 978-5-97060-969-9

© 2020 by Morgan & Claypool
© Оформление, издание, перевод, ДМК Пресс, 2021

Contents

Об авторах	10
Предисловие от издательства	13
Предисловие ко второму изданию	14
Глава 1. Знакомство с беспилотными технологиями	17
1.1. Обзор технологий в составе БТС	17
1.2. Алгоритмы БТС	18
1.2.1. Сбор данных об окружающей среде (сенсорное сканирование)	18
1.2.2. Восприятие	19
1.2.3. Распознавание и отслеживание объектов	22
1.2.4. Действия	22
1.3. Клиентская система БТС	24
1.3.1. Операционная система для роботов (ROS)	24
1.3.2. Аппаратная платформа	26
1.4. Облачная платформа БТС	27
1.4.1. Симуляция	27
1.4.2. Создание HD-карт	28
1.5. Это только начало	29
Глава 2. Локализация БТС	30
2.1. Локализация с помощью спутниковых навигационных систем	30
2.1.1. GNSS	30
2.1.1. Анализ погрешностей GNSS	31
2.1.3. Спутниковые системы дифференциальной коррекции	33
2.1.4. Кинематический и дифференциальный режимы GPS	33
2.1.5. Позиционирование высокой точности	35
2.1.6. Интеграция GNSS и INS	37
2.2. Локализация с помощью лидара и HD-карт	38
2.2.1. Технология LiDAR	38
2.2.2. HD-карты. Краткий обзор	41
2.2.3. Локализация посредством лидара и HD-карт	45
2.3. Визуальная одометрия	49
2.3.1. Стереовизуальная одометрия	50
2.3.2. Монокулярная визуальная одометрия	51
2.3.3. Инерциальная визуальная одометрия	52

2.4. Счисление координат и колесная одометрия	53
2.4.1. Колесные энкодеры.....	54
2.4.2. Погрешности колесной одометрии	55
2.4.3. Минимизация погрешностей колесной одометрии.....	56
2.5. Сбор и обобщение данных нескольких сенсоров	58
2.5.1. Беспилотный автомобиль Boss для городских условий от Университета Карнеги–Меллона (CMU).....	58
2.5.2. Беспилотный автомобиль Junior для городских условий от Стэнфордского университета.....	60
2.5.3. Bertha от Mercedes Benz	62
2.6. Справочные материалы.....	64

Глава 3. Алгоритмы восприятия беспилотных технологий

3.1. Введение	67
3.2. Датасеты	68
3.3. Обнаружение	70
3.4. Сегментация.....	72
3.5. Стереозрение, оптический поток, поток сцен	74
3.5.1. Стереоскопия и глубина	74
3.5.2. Оптический поток.....	75
3.5.3. Поток сцен	76
3.6. Отслеживание.....	77
3.7. Заключение	80
3.8. Справочные материалы.....	80

Глава 4. Глубокое обучение в алгоритмах восприятия беспилотных технологий

4.1. Сверточные нейронные сети	83
4.2. Обнаружение	84
4.3. Семантическая сегментация.....	88
4.4. Стереозрение и оптический поток	91
4.4.1. Стереозрение.....	91
4.4.2. Оптический поток.....	93
4.4.3. Обучение без учителя для плотного сопоставления изображений. 95	
4.5. Заключение	97
4.6. Справочные материалы.....	98

Глава 5. Прогнозирование и маршрутизация

5.1. Планирование и управление. Общие сведения	100
5.1.1. Архитектура: планирование и управление в широком смысле	100
5.1.2. Область ответственности каждого модуля: какие задачи решают модули	102
5.2. Прогнозирование движения	105

5.2.1. Поведенческое прогнозирование с точки зрения классификации объектов	107
5.2.2. Генерация траектории транспортного средства	111
5.3. Маршрутизация на основе полос движения	114
5.3.1. Применение взвешенного ориентированного графа для задач маршрутизации	115
5.3.2. Распространенные алгоритмы маршрутизации	118
5.3.3. Стоимость графа маршрутизации: слабая или сильная маршрутизация	121
5.4. Заключение	122
5.5. Справочные материалы	122

Глава 6. Принятие решений, планирование и управление

6.1. Принятие поведенческих решений	124
6.1.1. Подход с применением марковского процесса принятия решений	126
6.1.2. Подход «разделяй и властвуй» на основе сценариев	129
6.2. Планирование движения	137
6.2.1. Модель транспортного средства, модель дороги и система координат SL	140
6.2.2. Планирование движения на основе планирования пути и скорости	141
6.2.3. Планирование движения посредством продольного и поперечного планирования	148
6.3. Управление на основе обратной связи	152
6.3.1. Модель «Велосипед»	153
6.3.2. ПИД-регулирование	155
6.4. Заключение	156
6.5. Справочные материалы	157

Глава 7. Планирование и управление на основе обучения

с подкреплением	159
7.1. Введение	159
7.2. Обучение с подкреплением	161
7.2.1. Q-обучение	164
7.2.2. Методы «субъект–критик»	168
7.3. Планирование и управление БА на основе алгоритмов с обучением	170
7.3.1. Применение обучения с подкреплением на уровне принятия поведенческих решений	171
7.3.2. Применение обучения с подкреплением на уровнях планирования и управления	171
7.4. Заключение	175
7.5. Справочные материалы	175

Глава 8. Клиентские системы для беспилотных технологий	177
8.1. Сложность системы беспилотного управления	177
8.2. Операционная система для беспилотной технологии	179
8.2.1. Обзор ROS	179
8.2.2. Надежность системы.....	181
8.2.3. Повышение производительности.....	183
8.2.4. Управление ресурсами и безопасность	183
8.3. Вычислительная платформа	184
8.3.1. Реализация вычислительной платформы.....	184
8.3.2. Существующие вычислительные решения	184
8.3.3. Вопросы архитектуры вычислительной системы	186
8.4. Справочные материалы.....	189
Глава 9. Облачные платформы для беспилотных технологий	190
9.1. Введение	190
9.2. Инфраструктура	191
9.2.1. Структура распределенных вычислений	192
9.2.2. Распределенное хранение	192
9.2.3. Гетерогенные вычисления	193
9.3. Симуляция	195
9.3.1. BinPipeRDD	196
9.3.2. Объединение Spark и ROS.....	197
9.3.3. Производительность.....	197
9.4. Обучение модели	198
9.4.1. Почему именно Spark?.....	198
9.4.2. Архитектура обучающей платформы	199
9.4.3. Гетерогенные вычисления	200
9.5. Создание HD-карт	201
9.5.1. HD-карты	202
9.5.2. Создание карт в облаке.....	203
9.6. Заключение	204
9.7. Справочные материалы	205
Глава 10. Беспилотная доставка «до двери»	
в сложных дорожных условиях	206
10.1. Предыстория и предпосылки	206
10.2. Технологии беспилотной доставки в сложных дорожных условиях	207
10.3. JD.com: решение с использованием беспилотных технологий	209
10.3.1. Архитектура беспилотных технологий.....	209
10.3.2. Локализация и HD-карты	210
10.3.3. Восприятие	212
10.3.4. Прогнозирование, принятие решений и планирование движения.....	214

10.4. Стратегии защиты и обеспечения безопасности.....	217
10.4.1. Проверка в симуляции.....	217
10.4.2. Мониторинг на стороне транспортного средства	218
10.4.3. Удаленный мониторинг.....	218
10.5. Развертывание в производственной среде	218
10.6. Полученные уроки	219
10.7. Справочные материалы	220
Глава 11. Беспилотные транспортные средства от PerceptIn	221
11.1. Введение	221
11.2. Дорогостоящие технологии.....	222
11.2.1. Сенсорное сканирование.....	222
11.2.2. Локализация	223
11.2.3. Восприятие	223
11.2.4. Принятие решений	224
11.2.5. Создание и обслуживание HD-карт	224
11.2.6. Интеграция системы.....	225
11.3. Достижение ценовой доступности и надежности	225
11.3.1. Сбор и обобщение данных нескольких сенсоров	226
11.3.2. Модульная конструкция	227
11.3.3. Высокоточные визуальные карты.....	228
11.4. Развертывание сети беспилотных низкоскоростных электромобилей (LSEV) для предоставления услуг.....	229
11.5. Выводы.....	230
11.6. Справочные материалы.....	230
Применение этой книги в преподавании и обучении	232
1. Введение	232
2. Существующие учебные курсы по управлению БТС.....	233
3. Модульный и интегрированный подход к обучению	234
3.1. Методика преподавания	234
3.2. Учебные материалы к модулям	235
3.3. Интеграционные экспериментальные платформы	236
4. Пилотные тематические курсы.....	236
4.1. Введение в курс БТС	237
4.2. Изучение БТС как дополнение курса о встроенных системах.....	238
4.3. Курс профессиональной подготовки	238
5. Заключение	239
Справочные материалы	240

Об авторах



Д-р Шаошань Лю (Dr. Shaoshan Liu) является основателем и генеральным директором PerceptIn, компании, занимающейся разработкой беспилотных технологий. С момента основания PerceptIn получила финансирование на сумму более 12 миллионов долларов США от ведущих венчурных компаний, таких как Walden International, Matrix Partners и Samsung Ventures. До основания PerceptIn, Д-р. Шаошань Лю посвятил более 10 лет работе в ведущих научно-исследовательских институтах, среди которых Baidu USA, LinkedIn, Microsoft, Microsoft Research, INRIA, Intel Research и Broadcom.

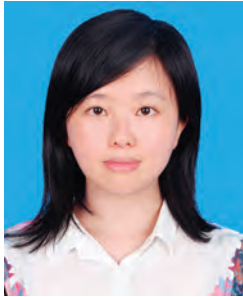
Д-р Шаошань Лю получил степень доктора компьютерной инженерии в Калифорнийском университете в Ирвайне. Он является автором более чем 60 исследовательских трудов, а также обладателем более 150 международных патентов США в области робототехники и беспилотных технологий. Также, он является ведущим автором популярной серии пособий «Создание беспилотных транспортных систем» (*Creating Autonomous Vehicle Systems*) и «Разработка беспилотных транспортных средств и роботов» (*Engineering Autonomous Vehicles and Robots*). Д-р. Шаошань Лю – старший член ассоциации Института инженеров электротехники и электроники (IEEE – Institute of Electrical and Electronics Engineers), а также выдающийся спикер компьютерного общества IEEE (IEEE Computer Society) и Ассоциации вычислительной техники (ACM – Association for Computing Machinery). Также, он является основателем Специального технического сообщества IEEE (IEEE Special Technical Community) в направлении разработки беспилотных транспортных систем. Эл. адрес: shaoshan.liu@perceptin.io.



Д-р Лиюнь Ли (Dr. Liyun Li) имеет более чем 6-летний опыт разработки ПО для беспилотных технологий. В настоящее время он является главным инженером и менеджером в Xpeng Motors (NYSE: XPEV), где он руководит разработкой программного обеспечения для Navigation Guided Pilot (NGP). До прихода в Xpeng Motors он занимал должность главного инженера в JD.com. Он является одним из основателей команды по разработке беспилотных технологий Baidu USA, где он руководил работой по созданию основных модулей в беспилотной системе Baidu с открытым исходным кодом, в частности модулей

планирования и прогнозирования. Д-р Ли является автором двух книг в области беспилотных технологий, а именно «Создание беспилотных транспортных

систем» (Creating Autonomous Vehicle Systems) (Morgan & Claypool Publishers) и «Первая книга о беспилотных технологиях» (The First Technology Book in Autonomous Driving) (Publishing House of Electronics Industry (PHEI)). Также, он является обладателем более чем 20 международных патентов в области беспилотных технологий. Д-р Ли получил степень доктора компьютерных наук в Нью-Йоркском университете (New York University) и степень бакалавра электронной инженерии в Университете Цинхуа (Tsinghua University).



Д-р Цзе Тан (Dr. Jie Tang) является действующим доцентом факультета компьютерных наук и инженерии Южно-Китайского технологического университета (Гуанчжоу, Китай) (South China University of Technology). Ранее она работала в качестве научного сотрудника в Калифорнийском университете (Риверсайд) (University of California) и в Университете Кларксона (Clarkson University) с декабря 2013 по август 2015 г. В 2006 г. Д-р Тан получила степень бакалавра по направлению компьютерных наук в Оборонном научно-техническом университете. В 2012 г. она получила степень доктора

компьютерных наук в Пекинском технологическом институте (Beijing Institute of Technology). С 2009 по 2011 год она была приглашенным исследователем в Лаборатории параллельных систем и компьютерной архитектуры (PASCAL – PArallel Systems and Computer Architecture Lab) Калифорнийского университета в Ирвайне, США. Эл. адрес: cstangjie@scut.edu.cn.



Д-р Шуан Ву (Dr. Shuang Wu) является действующим научным сотрудником Yitu Technology. Ранее он работал старшим научным сотрудником лаборатории искусственного интеллекта Baidu (Baidu's AI lab) в Саннивейле, Калифорния, и старшим архитектором в Baidu USDC. Он получил степень доктора физики в Университете Южной Калифорнии (University of Southern California) и некоторое время работал в качестве молодого научного сотрудника в Калифорнийском университете в Лос-Анджелесе (UCLA – University of California, Los Angeles). Он занимался исследованиями в области компьютерного и биологического зрения, а также изучал прикладное

машинное обучение и его применение в промышленности для компьютерной рекламы и распознавания речи. Его труды были опубликованы такими конференциями как NIPS (Neural Information Processing Systems) и ICML (International Conference on Machine Learning).



Д-р Жан-Люк Годье (Dr. Jean-Luc Gaudiot) получил диплом инженера (Diplôme d'Ingénieur) в ESIEE (Париж, Франция) в 1976 г. После он получил степени магистра и доктора компьютерных наук Калифорнийском университете в Лос-Анджелесе в 1977 и 1982 г. соответственно. В настоящее время он является профессором кафедры электротехники и информатики в Калифорнийском университете в Ирвайне (UCI – University of California, Irvine). До прихода в UCI в 2002 г., с 1982 г. он был профессором электротехники в Университете Южной Калифорнии (University of Southern California). В область его

научных интересов входят многопоточные архитектуры, отказоустойчивые мультипроцессоры, а также реализация реконфигурируемых архитектуры. На его счету более 250 опубликованных статей для журналов и конференций. Спонсорами его исследований были NSF, DoE и DARPA, а также ряд других промышленных компаний. Д-р Жан-Люк Годье находился на самых разных должностях и в 2017 г. был избран президентом компьютерного сообщества IEEE (IEEE Computer Society). Эл. адрес: gaudiot@uci.edu.

Предисловие ко второму изданию

На сегодняшний день беспилотные транспортные средства (БТС) – наземные, водные или воздушные – уже вошли в нашу жизнь и применяются в самых разных областях, начиная от сервисов беспилотного такси и заканчивая автоматическими системами воздушного наблюдения за особо важными удаленными районами. Благодаря непрерывному техническому прогрессу за последние несколько десятилетий человечество смогло реализовать подобные инновации. Однако на сегодняшний день в этой области существует ряд проблем, связанных с проектированием. Так, задача исследователей и инженеров заключается в преодолении этих проблем, что позволит обеспечить продуктивное, эффективное и, что особенно важно, безопасное функционирование описанных выше многочисленных и сложных автономных устройств.

Таким образом, цель этой книги – предоставить обзор проблем проектирования беспилотных транспортных систем, а также познакомить читателя с некоторыми общими проектными решениями. Обязательными условиями успешной разработки беспилотных систем являются хорошее технологическое оснащение, интеграция оборудования и программного обеспечения, а также синергия с резидентными платформами (такими как облачные серверы). Основное внимание в этой книге уделяется наземным транспортным средствам и, в частности, автомобилям и их функционированию в городских и внегородских условиях, а также в условиях бездорожья. Эта книга адресована инженерам, работающим как в академической, так и в промышленной областях. Она предлагает обзор и решение текущих проблем, а также проблем, которые могут возникнуть при будущих исследованиях. Среди них могут оказаться как проблемы сенсорного сканирования и восприятия у автономных систем, так и проблемы иного плана, например поддержка облачных серверов. Полезным дополнением к книге является большое количество библиографических ссылок, которые помогут читателю сориентироваться в «дебрях» предыдущих работ.

Аудитория тепло приняла первое издание этой книги. Многие университеты включили ее в программы своих курсов обучения, а также она послужила материалом для внутренних тренингов в некоторых организациях (более подробную информацию об этом можно найти в новом разделе «Преподавание и обучение»). Кроме того, первое издание получило множество полезных комментариев и отзывов от читателей, которые значительно повлияли на совершенствования содержания второго издания. В частности, на основе этих комментариев и отзывов мы внесли следующие изменения: во-первых, мы расширили главы 3 и 4 и добавили в них информацию о передовых методах восприятия; во-вторых, мы переработали главы 5, 6 и 7, что позволило доступнее донести содержащуюся в них информацию до читателей; в-третьих, в главы 10 и 11 мы добавили примеры промышленных испытаний систем, которые позволят читателям применять полученные знания на практике. Помимо

этого, для преподавателей мы добавили раздел под названием «Преподавание и обучение», который призван содействовать при использовании этой книги в программе обучающих курсов. Мы считаем, что преимущество второго издания состоит в том, что наряду с информацией о последних достижениях в области автономных транспортных систем оно содержит и практические примеры из реальной жизни, которые позволят читателям успешно применять полученные знания на практике.

СТРУКТУРА КНИГИ

В *главе 1* представлена краткая история информационных технологий, а также обзор алгоритмов, лежащих в основе автономных транспортных систем, архитектуры таких систем и необходимой инфраструктуры поддержки. В *главе 2* представлены наиболее популярные подходы к локализации, которая является одной из наиболее важных задач области. Также здесь подробно описаны преимущества, недостатки и особенности интеграции различных версий принципов *глобальной спутниковой навигационной системы (GNSS – Global Navigation Satellite System)*, *инерциальной навигационной системы (INS – Inertial Navigation system)*, а также принципов работы *лидара (LiDAR – Light Detection and Ranging)* и колесной одометрии. В *главе 3* описаны принципы восприятия («понимания») окружающей среды на основе данных сенсорного сканирования, а также представлены обзорные описания различных прикладных алгоритмов, таких как интерпретация сцен, обработка потоков изображений, отслеживание и др. В *главе 4* представлены подходы глубокого обучения к восприятию, основанные на обработке больших наборов данных и сложных вычислениях, необходимых для классификации изображений, обнаружения объектов, семантической сегментации и многого другого. *Глава 5* описывает задачи БТС, возникающие после исследования окружающей среды, такие как прогнозирование будущих событий (например, движения другого транспортного средства, находящегося неподалеку). Далее, в *главе 6*, подробно описывается процессы принятия решений, планирования и контроля. Обратная связь между модулями, в том числе и принятие ими независимых друг от друга решений, а также разрешение конфликтов (например, один модуль может рекомендовать смену полосы движения, а другой может обнаружить препятствие на этой полосе) рассматривается с упором на описание алгоритмов принятия поведенческих решений (таких как *Марковский процесс принятия решений [MDP – Markov decision process]*, основанный на принципе «разделяй и властвуй») и планирования движения. Все это подводит нас к *главе 7*, где говорится о необходимости реализации алгоритмов планирования и контроля, основанных на обучении с подкреплением, для полной интеграции ситуационных сценариев при разработке автономной системы. *Глава 8* содержит информацию о бортовой вычислительной платформе. Здесь вы можете найти общее описание экосистемы под названием *операционная система для роботов (ROS – Robot Operating System)*, а также описание реально используемого оборудования. Рассказывая о потребности в гетерогенных вычислениях, мы также подчеркиваем необходимость умения автономных систем производить вычисления в режиме реального времени (обработка данных с сенсоров), а также рассматриваем некоторые приклад-

ные аспекты (энергопотребление и рассеивание тепла). Таким образом, делаем вывод, что для создания автономных систем требуется использование разных обрабатывающих модулей на основе *центральных процессоров* (CPU – Central Processing Unit) общего назначения, *графических процессоров* (GPU – Graphic Processing Unit), *программируемых пользователем вентильных матриц* (FPGA – Field-Programmable Gate Array) и т. д. В *главе 9* рассматривается инфраструктура облачной платформы, которая служит для «связывания всех модулей воедино» (т. е. обеспечивает распределение имитационных тестов для новых алгоритмов, автономного обучения модели глубокого обучения и построения карт высокой четкости (HD-карт). *Глава 10* представляет учебный пример коммерческого беспилотного автомобиля (БА), предназначенного для доставки «до двери» в сложных дорожных условиях. В *главе 11* представлен анализ доступных на сегодняшний день БТС, используемых для оказания услуг микротранзитных перевозок.

Глава 1

Знакомство с беспилотными технологиями

Сегодня мы стоим на пороге рассвета беспилотных технологий. Чтобы понять, что нас ожидает в будущем, необходимо обратиться к истории, с чего мы и начнем.

Начало развития информационных технологий приходится на 1960-е, когда Fairchild Semiconductors и Intel заложили их фундамент, выпустив первые в мире микропроцессоры, а в качестве побочного продукта основав Кремниевую долину. Однако, несмотря на то что микропроцессорные технологии обеспечивали значительный прирост производительности, они были недоступны для широкого применения. В 1980-х годах Microsoft и Apple заложили фундамент второго этапа развития информационных технологий, представив графический пользовательский интерфейс и реализовав концепцию «ПК/Мас в каждом доме». В 2000-х интернет-компании под эгидой Google дали старт третьему этапу, связав людей и информацию, т. е. обеспечили потенциальную связь между источниками и потребителями информации. Затем в 2010-х создатели социальных сетей, таких как Facebook и LinkedIn, положили начало четвертому этапу, буквально переместив человеческое общество в интернет и позволив людям напрямую взаимодействовать друг с другом. По мере значительного увеличения количества интернет-пользователей примерно в 2015 году появились Uber и Airbnb, которые начали пятый этап развития информационных технологий и сформировали коммерческие интернет-площадки. Однако, несмотря на то что Uber и Airbnb предоставили пользователям средства для доступа к поставщикам услуг через интернет, услуги по-прежнему предоставлялись людьми.

1.1. ОБЗОР ТЕХНОЛОГИЙ В СОСТАВЕ БТС

Как представлено на рис. 1.1, БТС представляет собой не одну технологию, а сложную систему технологий, состоящую из множества подсистем. Итак, разобьем их на три основные группы: алгоритмы сенсорного сканирования, восприятия и принятия решений (требующие, как правило, построения сложных логических выводов); клиентские системы, включая операционную систему и аппаратную платформу; и облачные платформы, в том числе создание карт высокой четкости (HD-карт), обучение моделей глубокого обучения, моделирование и хранение данных.

Первая подсистема отвечает за извлечение важной информации из необработанных данных, полученных сенсорами, и обеспечивает исследование роботом окружающей среды, на чем в дальнейшем строится принятие решений относительно его будущих действий. Клиентские системы объединяют алгоритмы первой подсистемы, тем самым позволяя им работать в режиме реального времени и обеспечивая надежность. Например, если камера генерирует данные с частотой 60 Гц, клиентским системам необходимо убедиться, что самый длинный этап обработки занимает менее 16 мс. Платформа облачных вычислений отвечает за автономные вычисления и хранение данных. С ее помощью мы можем тестировать новые алгоритмы, обновлять HD-карты и обучать БА более качественным моделям распознавания, отслеживания и принятия решений.

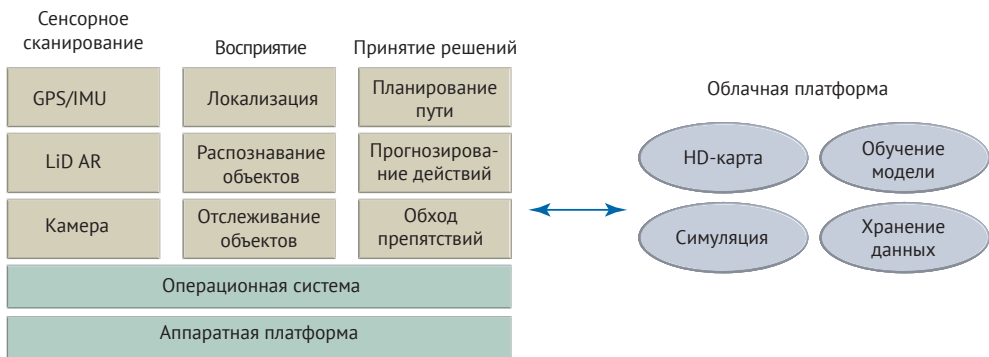


Рис. 1.1. Обзор архитектуры БТС

1.2. Алгоритмы БТС

Алгоритмы БТС включают в себя: сенсорное сканирование окружающей среды, т. е. извлечение важной информации из необработанных данных, полученных датчиками; восприятие, сущность которого заключается в локализации транспортного средства и исследовании окружающей среды; и принятие решений, другими словами, принятие мер, обеспечивающих надежное и безопасное достижение транспортным средством пункта назначения.

1.2.1. Сбор данных об окружающей среде (сенсорное сканирование)¹

Обычно на борту БА располагается несколько основных сенсоров. Каждый из них имеет свои преимущества и недостатки, поэтому – из соображений надежности и безопасности – сенсоры работают в совокупности. Рассмотрим их виды:

- *GPS/IMU*: система GPS/IMU обеспечивает локализацию (вычисление пространственных координат) БТС, сообщая данные инерциальной си-

¹ На английском языке процесс сбора информации при помощи сенсоров называется *sensing*, однако устоявшегося русского перевода у этого термина нет, поэтому в данной книге будет применяться перевод «сенсорное сканирование», так как он лучше всего отражает суть циклического сбора данных с набора сенсоров. – Прим. перев.

стемы и глобальную оценку местоположения с высокой частотой (примерно 200 Гц). GPS – довольно точная система, но она имеет низкую частоту обновления (всего около 10 Гц) и поэтому не может предоставлять информацию в режиме реального времени. Что касается IMU, у таких сенсоров наблюдается тенденция накопления погрешностей, что впоследствии приводит к ухудшению оценок местоположения. Но зато IMU может предоставлять обновления чаще, с частотой 200 Гц или выше. Комбинируя GPS и IMU, мы можем предоставлять точные данные о местоположении транспортных средств в реальном времени;

- *LiDAR*: лидар используется для обзора местности, определения местоположения и обхода препятствий. Способ его работы заключается в том, что за счет отражения луча от окружающих предметов он измеряет время прохождения луча и вычисляет расстояния до них. Благодаря высокой точности лидар применяется для создания HD-карт, определения местоположения движущегося транспортного средства на HD-картах, обнаружения препятствий и многого другого. Обычно блок лидара, такой как 64-лучевой лазер Velodyne, вращается с частотой 10 об/мин и осуществляет около 1,3 млн считываний в секунду;
- *камеры*: в основном камеры используются для отслеживания и распознавания объектов, а также для решения задач типа выбора полосы движения, обнаружения светофоров, пешеходов и т. д. В существующих реализациях, как правило, на БА по кругу устанавливают восемь или более камер с разрешением 1080p. Таким образом, появляется возможность обнаружения, распознавания и отслеживания объектов спереди, сзади и по обе стороны БА. Обычно камеры работают с частотой 60 Гц и в совокупности генерируют около 1,8 Гб необработанных данных в секунду;
- *радар и сонар*: радар и сонар используются в качестве последней линии обороны при уклонении от препятствий. Данные, получаемые этими датчиками, показывают расстояние до ближайшего объекта, а также скорость транспортного средства. Как только БА обнаруживает, что приближается к какому-либо объекту и возникает опасность столкновения, он тормозит или поворачивает, чтобы избежать препятствия. Следовательно, данные, генерируемые радаром и сонаром, практически не требуют обработки и обычно поступают непосредственно в процессор управления, что позволяет реализовать такие «экстренные» функции, как поворот, торможение или предварительное натяжение ремней безопасности.

1.2.2. Восприятие

Далее данные с сенсоров передаются на стадию восприятия. Три основных задачи этапа восприятия – это локализация, обнаружение и отслеживание объекта.

Для локализации можно использовать связку GPS/IMU, и, как упоминалось выше, GPS обеспечивает довольно точную локализацию, но со сравнительно низкой частотой обновления, в то время как IMU предоставляет моментальные обновления за счет менее точных данных. Для объединения преимуществ двух сенсоров обычно используют *фильтр Калмана*. Суть этого метода представлена на рис. 1.2: IMU обновляет положение автомобиля каждые 5 мс, но со временем накапливаются погрешности. Каждые 100 мс приходят обновления от GPS, которые помогают исправить погрешности IMU. Благодаря

этому методу комбинация GPS/IMU может обеспечивать быструю и точную локализацию БА. Тем не менее мы не можем полагаться исключительно на эту комбинацию по трем причинам: (1) сенсоры могут работать с необходимой точностью лишь в пределах одного метра; (2) сигнал GPS имеет проблемы с многолучевым распространением, поэтому он может отражаться от зданий, создавая помехи; и (3) GPS требует беспрепятственной передачи со спутников и, следовательно, не будет получать данные в туннелях или подобных местах.

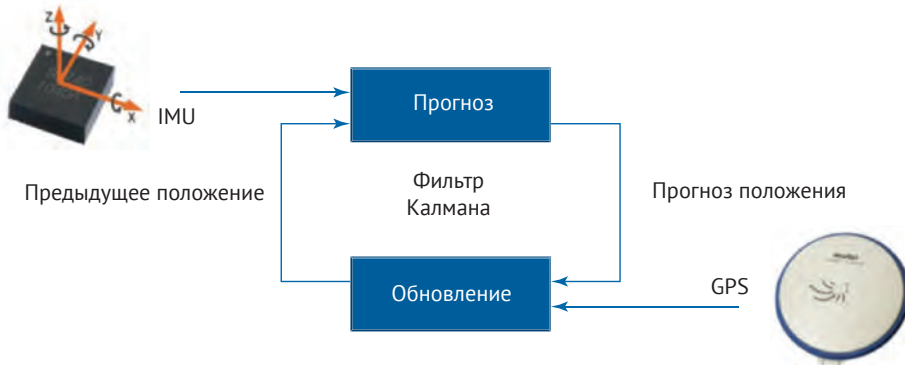


Рис. 1.2. Локализация при помощи GPS/IMU

Также для локализации можно использовать данные камер. Локализация с помощью камер может быть реализована в виде упрощенной схемы: (1) сначала путем триангуляции пар стереоизображений мы получаем *карту диспаратности*² (disparity map), которую можно использовать для получения информации о глубине для каждой точки; (2) сопоставив характерные признаки между последовательными кадрами стереоизображения, мы можем установить корреляции между ними в разных кадрах. Затем можем оценить перемещение БТС за время, прошедшее между двумя прошлыми кадрами; а также, (3) сравнивая характерные особенности с теми, что есть на известной карте, мы можем получить данные о текущем положении транспортного средства. Все же такой подход к локализации очень чувствителен к условиям освещения и, следовательно, не является надежным.

Именно поэтому лидар использует методы *фильтра частиц*. Облако точек, генерируемое лидаром, «описывает форму» окружающей среды, но отдельные точки датчик различает с трудом. Используя фильтр частиц, система сравнивает конкретную наблюдаемую форму облака точек с уже имеющейся картой. Для определения местонахождения движущегося транспортного средства относительно этих карт мы можем применить метод фильтра частиц, который позволяет сопоставить данные лидара с картой. Мы выяснили, что метод фильтрации частиц обеспечивает локализацию в реальном времени с точностью до 10 см и эффективен в городских условиях. Однако у лидара есть одна проблема: когда в воздухе много взвешенных частиц, таких как капли дождя

² Карта смещений/расхождений между двумя изображениями. – Прим. ред.

и пыль, точность измерений снижается за счет возникающего шума. Следовательно, как представлено на рис. 1.4, для достижения надежной и точной локализации нам необходимо использовать все датчики в совокупности, объединяя их преимущества.

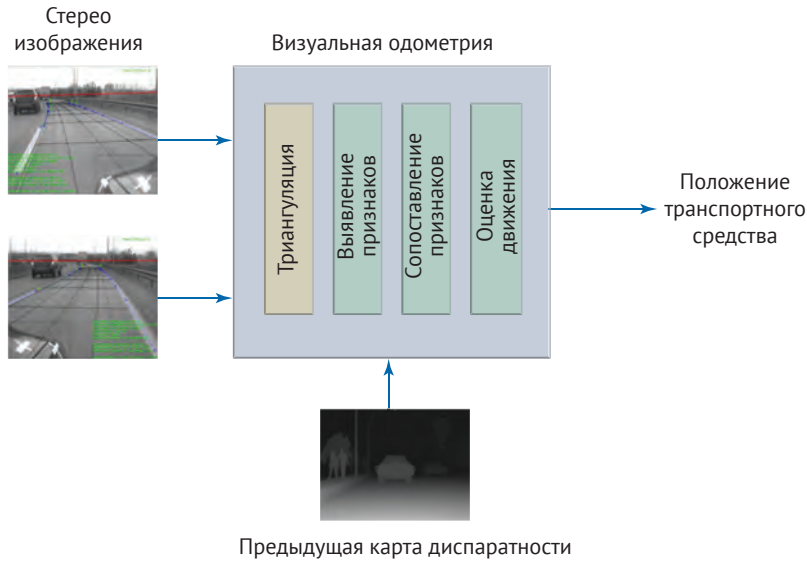


Рис. 1.3. Стереовизуальная одометрия

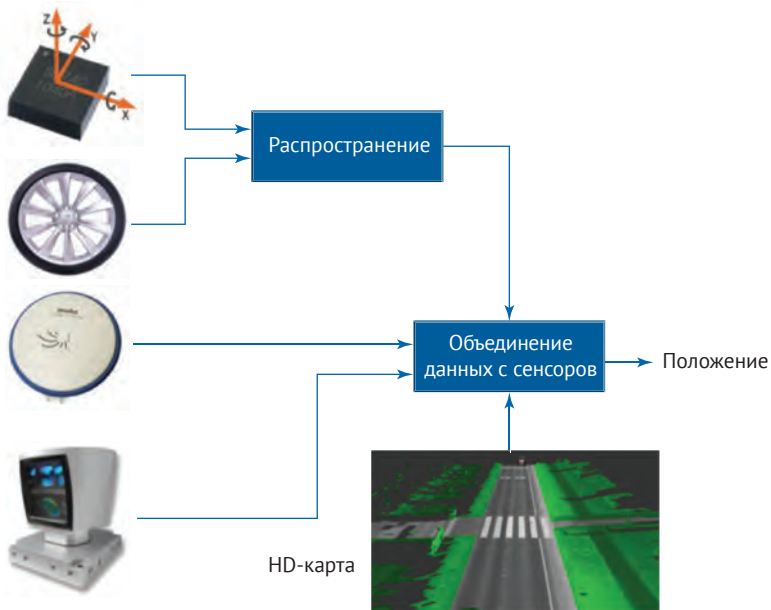


Рис. 1.4. Процесс локализации с помощью объединения сенсоров

1.2.3. Распознавание и отслеживание объектов

Первоначально для обнаружения и отслеживания объектов использовался лидар, поскольку он способен предоставлять точную информацию о глубине. В последние годы мы стали свидетелями быстрого развития технологии глубокого обучения, которая позволяет относительно точно обнаруживать и отслеживать объекты. *Сверточная нейронная сеть (CNN – Convolution Neural Network)* – это тип *глубокой нейронной сети (DNN – Deep Neural Network)*, которая широко используется для распознавания объектов. CNN обычно состоит из нескольких слоев. (1) *Сверточный слой*, используя разные фильтры, извлекает различные признаки из образов входных данных. Каждый фильтр содержит набор «обучаемых» параметров. (2) *Слой активации* принимает решение об активации целевого нейрона. (3) *Объединяющий слой* уменьшает пространственный размер представления, чтобы уменьшить количество параметров и, следовательно, вычислений в сети. (4) *Полностью подключенный слой* характерен тем, что нейроны устанавливают полные связи с активированными нейронами предыдущего слоя.

Отслеживание объектов относится к автоматической оценке траектории объекта во время его движения. После того как отслеживаемый объект идентифицирован с использованием методов распознавания объектов, цель отслеживания объекта состоит в том, чтобы впоследствии автоматически отслеживать траекторию объекта. Такая технология может использоваться для отслеживания ближайших движущихся транспортных средств, а также людей, переходящих дорогу, чтобы предотвратить столкновение БА с ними. В последние годы методы глубокого обучения демонстрировали преимущества в отслеживании объектов в сравнении с традиционными методами. В частности, с помощью вспомогательных естественных изображений *составной автокодировщик* можно обучать в автономном режиме, что позволит изучить общие признаки изображения, которые более устойчивы к вариациям точек обзора и положениям транспортных средств. Затем автономную обученную модель можно применять для онлайн-трекинга.

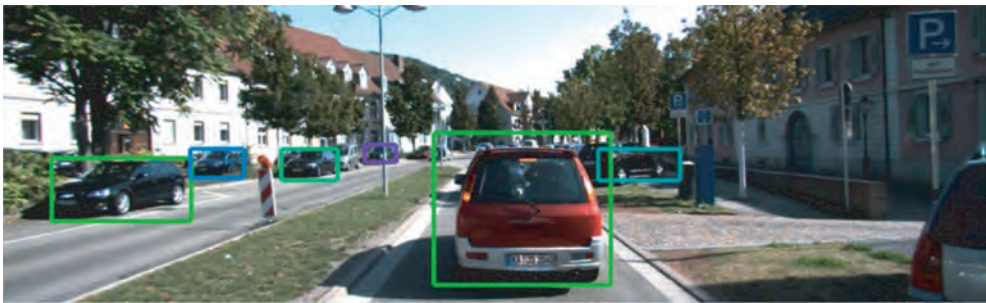


Рис. 1.5. Распознавание и отслеживание объектов [34] (используется с разрешения правообладателя)

1.2.4. Действия

Основываясь на исследовании среды, в которой находится БА, этап принятия решения обеспечивает безопасный и эффективный план действий в режиме реального времени.

Прогноз действий

Одна из основных задач для водителей-людей при перемещении в транспортном потоке – наблюдать и оценивать действия других водителей, которые напрямую влияют на собственную стратегию вождения. Особенно актуально это на дорогах с несколькими полосами движения, а также в моменты, когда автомобиль находится в точке смены движения. Чтобы убедиться в безопасности передвижений, блок принятия решений БА генерирует прогнозы действий ближайших транспортных средств и принимает решения на основе этих прогнозов. Чтобы предсказать действия других транспортных средств, можно сгенерировать стохастическую (вероятностную) модель наборов достижимых положений других участников движения и связать эти наборы достижимости с распределением вероятностей.

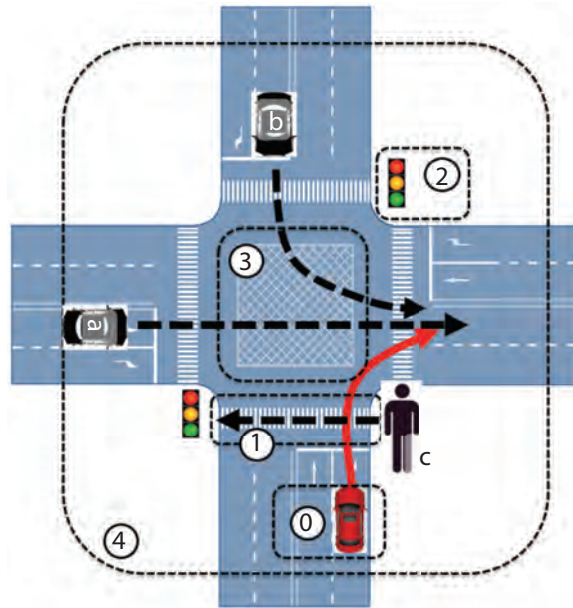


Рис. 1.6. Прогноз действий

Планирование пути

Планирование пути маневренного БА в динамической среде представляет собой довольно сложную задачу, особенно в те моменты, когда от БА требуется использовать все возможности маневрирования. «Грубый» метод заключается в поиске всех возможных путей и использовании критерия качества для определения наилучшего пути. Однако он требует огромных вычислительных ресурсов и может быть неприменим в режиме реального времени. С целью обойти описанные сложности были созданы вероятностные планировщики, обеспечивающие эффективное планирование пути в реальном времени.

Обход препятствий

Поскольку безопасность является первостепенной задачей при разработке БТС, обычно создается как минимум два уровня механизмов обхода препятствий. Такой подход может гарантировать, что транспортное средство не столкнется с препятствиями. Первый уровень является проактивным и основан на прогнозировании трафика. Механизм прогнозирования трафика генерирует такие показатели, как время до столкновения или минимальное расстояние, и на основе этой информации запускается механизм обхода препятствий для выполнения локального перепланирования пути. Если данный механизм выйдет из строя, то задачу возьмет на себя второй уровень – механизм реагирования, использующий данные радара. Как только радар обнаруживает препятствие, он запускает механизм замещающего управления, чтобы избежать столкновения.

1.3. КЛИЕНТСКАЯ СИСТЕМА БТС

Клиентские системы объединяют вышеупомянутые алгоритмы для удовлетворения требований функционирования в реальном времени и надежности. Основные сложности здесь заключаются в следующем: во-первых, системе необходимо обеспечить скорость обработки огромного количества генерируемых сенсорами данных; во-вторых, если часть системы выходит из строя, необходимо обеспечить гарантию восстановления после сбоя; и в-третьих, все вычисления должны выполняться в условиях ограниченной энергии и ресурсов БА.

1.3.1. Операционная система для роботов (ROS)

ROS – это широко используемая мощная среда распределенных вычислений, разработанная для приложений робототехники. На рис. 1.7 представлено, как каждая задача (например, локализация) размещается в узле ROS. Узлы ROS могут общаться друг с другом через темы и службы. ROS отлично подходит для управления БТС, за исключением нескольких проблем: (1) надежность: ROS имеет одно главное устройство и не обладает функциями мониторинга для восстановления отказавших узлов; (2) производительность: при отправке сообщений на широкоэвещательные каналы она многократно дублирует их, что приводит к снижению производительности; (3) безопасность: у системы нет механизмов аутентификации и шифрования. Решить эти проблемы была призвана версия ROS 2.0, однако ее тестирование еще не производилось, а многие ее функции еще недоступны. Следовательно, чтобы использовать ROS в управлении БТС, сначала необходимо найти решение всех вышеперечисленных проблем.

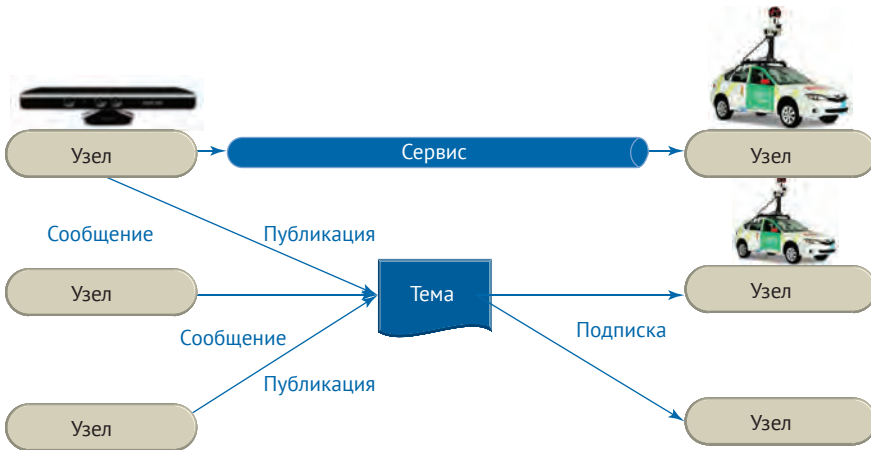


Рис. 1.7. Операционная система для роботов (ROS)

Надежность

Текущая реализация ROS имеет только один главный узел, поэтому, если он выйдет из строя, произойдет сбой всей системы. Разумеется, это не соответствует требованиям безопасности для БА. Чтобы решить эту проблему, в ROS был реализован механизм, похожий на ZooKeeper. На рис. 1.8 показано, что в этой схеме, помимо главного узла, есть и резервный. В случае отказа основного узла резервный узел возьмет на себя управление, что позволит системе продолжить работу. Кроме того, механизм ZooKeeper отслеживает и перезапускает все вышедшие из строя узлы, обеспечивая надежность всей системы ROS.

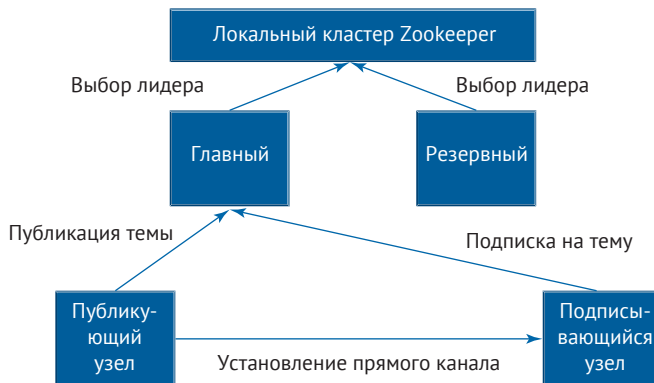


Рис. 1.8. Zookeeper для ROS

Производительность

Еще одна проблема текущей реализации ROS – производительность. Узлы ROS обмениваются данными очень часто, поэтому крайне важно обеспечить эффективную связь между ними. Во-первых, когда локальные узлы связываются друг с другом, коммуникация происходит через механизм обратной связи. Каждый раз добавляются накладные расходы в 20 мс. С целью устра-

нить накладные расходы используется механизм общей памяти. Так, теперь сообщению не нужно проходить через стек TCP/IP, чтобы добраться до узла назначения. Во-вторых, когда узел ROS транслирует сообщение, оно копируется несколько раз, значительно уменьшая пропускную способность системы. Как представлено на рис. 1.9, переключение на механизм широковещательной рассылки может значительно повысить пропускную способность системы.

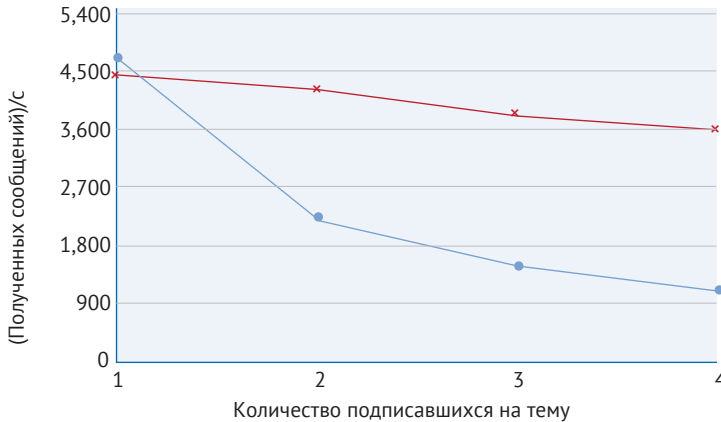


Рис. 1.9. Multicast и широковещательная рассылка в ROS

Безопасность

Как мы знаем, безопасность – это самый важный вопрос для ROS. Представьте два сценария: в первом сценарии узел ROS похищают (захватывают над ним контроль) и непрерывно посылают запросы на выделение памяти, пока в системе не заканчивается память и не начинается отключение других узлов ROS. В этом сценарии хакер успешно сможет создать сбой в системе. Во втором сценарии, поскольку по умолчанию сообщения ROS не зашифрованы, хакер легко перехватывает сообщения между узлами и применяет атаку типа «злонамеренный посредник». Чтобы решить первую проблему, мы можем использовать *контейнер Linux (LXC – Linux Container)*, чтобы ограничить количество ресурсов, используемых каждым узлом, а также предоставить механизм песочницы для защиты узлов друг от друга и тем самым эффективно предотвратить утечку ресурсов. Чтобы решить вторую проблему, мы можем зашифровать сообщения и предотвратить их перехват.

1.3.2. Аппаратная платформа

Чтобы разобраться в проблемах, возникающих при разработке аппаратной платформы для БТС, следует рассмотреть реализацию вычислительной платформы от ведущей компании в области БТС. Данная платформа состоит из двух вычислительных блоков, каждый из которых оснащен процессором Intel Xeon E5 и имеет от четырех до восьми графических ускорителей Nvidia K80. Второй вычислительный блок выполняет точно такие же задачи, что и первый, и используется для обеспечения надежности: в случае отказа первого блока второй может немедленно вступить в работу. Если же оба блока будут работать на пико-

вой производительности (более 5000 Вт потребляемой мощности), это приведет к сильному перегреву. Кроме того, каждый блок стоит 20 000–30 000 долл., что делает данное решение недоступным для среднестатистического потребителя.

Потребляемая мощность, количество выделяемого тепла и стоимость оборудования не позволяют поставить создание БА на поток. Чтобы исследовать границы диапазона и понять, насколько хорошо системы БТС могут работать на мобильной SoC³ ARM, мы реализовали упрощенную систему на мобильной SoC на базе ARM с пиковым энергопотреблением всего 15 Вт. Оказывается, производительность была близка к нашим требованиям: конвейер локализации был способен обрабатывать 25 изображений в секунду, почти не отставая от скорости генерации 30 изображений в секунду. Благодаря глубокому обучению появилась возможность выполнять 2-3 задачи распознавания объектов в секунду. Этап планирования и контроля позволил спланировать путь за 6 мс. С помощью этой системы мы могли вести машину со скоростью около пяти миль и не терять ее локализации.

1.4. ОБЛАЧНАЯ ПЛАТФОРМА БТС

БТС представляют собой мобильные системы, поэтому им требуется облачная платформа. Две основные функции облачной платформы заключаются в осуществлении распределенных вычислений и распределенном хранении. Облачные технологии имеют несколько приложений, таких как моделирование, которые используются для проверки новых алгоритмов. Также, они могут обеспечить создание HD-карт и обучение моделей глубокого обучения. При создании облачной платформы для распределенных вычислений мы использовали Spark, для гетерогенных вычислений – OpenCL и для хранения в памяти – Alluxio. Объединив вместе Spark, OpenCL и Alluxio, мы смогли создать надежное облако с низким показателем задержки и высокой пропускной способностью.

1.4.1. Симуляция

Основное применение такой системы – моделирование. Когда мы разрабатываем новый алгоритм, нужно тщательно протестировать его перед внедрением в БА. Тестировать алгоритмы на реальных БА было бы слишком дорого и затратно по времени, поэтому обычно мы тестируем новые алгоритмы на симуляторах, например с помощью воспроизведения данных через узлы ROS. Тестирование нового алгоритма даже на одном БА было бы слишком затратным. Как представлено на рис. 1.10, для решения этой проблемы мы разработали платформу распределенного моделирования. В данном случае для управления распределенными вычислительными узлами мы используем Spark, и на каждом узле мы запускаем экземпляр воспроизведения ROS. В тестовом наборе БТС для распознавания объектов, который мы использовали, потребовалось три часа для запуска на одном сервере; при использовании разработанной нами распределенной системы тест завершился за 25 мин (притом что мы увеличили количество автомобилей до восьми).

³ System-on-Crystall – система на кристалле, полнофункциональная система на одной микросхеме. – *Прим. ред.*

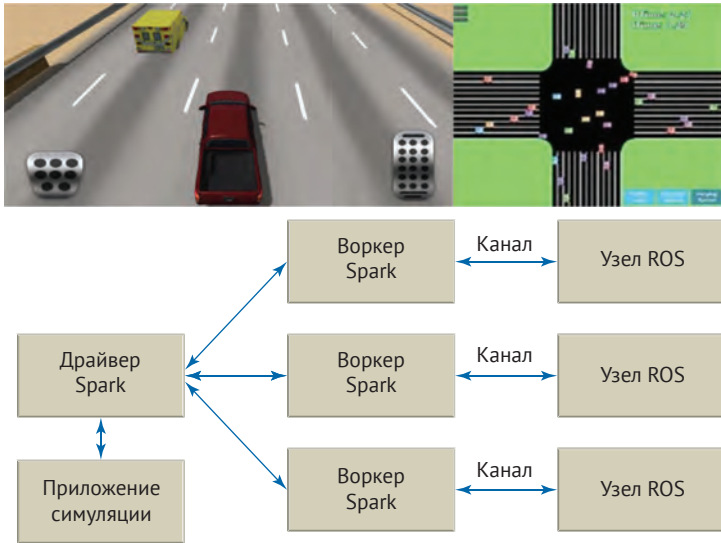
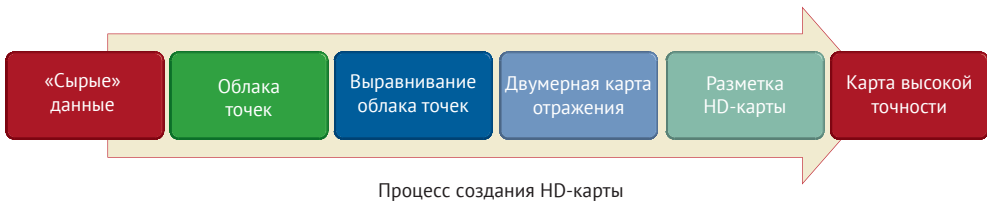


Рис. 1.10. Платформа моделирования на основе Spark и ROS

1.4.2. Создание HD-карт

На рис. 1.11 представлен процесс создания HD-карты. Как видите, этот процесс довольно сложен и включает в себя множество этапов, в том числе обработку данных, полученных датчиками, создание и выравнивание карты точек, создание 2D-карты отражательной способности, маркировку HD-карты, а также создание конечного варианта карты. Посредством Spark мы объединили все эти этапы в один. Что еще более важно, с помощью Spark можно реализовать вычислительный механизм непосредственно в памяти, так что теперь у нас нет необходимости хранить промежуточные данные на жестком диске, что значительно снижает затраты производительности в процессе создания карты.



Процесс создания HD-карты

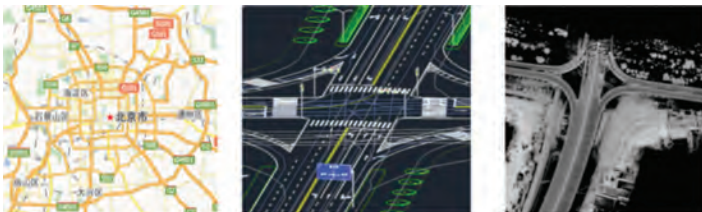


Рис. 1.11. Создание HD-карты на основе облака

Поскольку при создании БТС мы используем различные модели глубокого обучения, крайне важно предоставлять обновления для постоянного повышения эффективности и действенности этих моделей. Однако, поскольку объем сгенерированных необработанных данных огромен, мы не сможем добиться быстрого обучения модели с использованием отдельных серверов. Чтобы решить эту проблему, мы разработали хорошо масштабируемую распределенную систему глубокого обучения с использованием Spark и Paddle (платформа глубокого обучения, исходный код которой недавно был открыт Baidu). Как представлено на рис. 1.12, в драйвере Spark мы управляем контекстом Spark и контекстом Paddle, и в каждом узле исполнитель Spark размещает экземпляр тренера Paddle. Кроме того, в качестве сервера параметров для этой системы мы используем Alluxio. Такой подход позволил нам достичь линейного масштабирования производительности по мере добавления дополнительных ресурсов, что доказывает высокую масштабируемость системы.

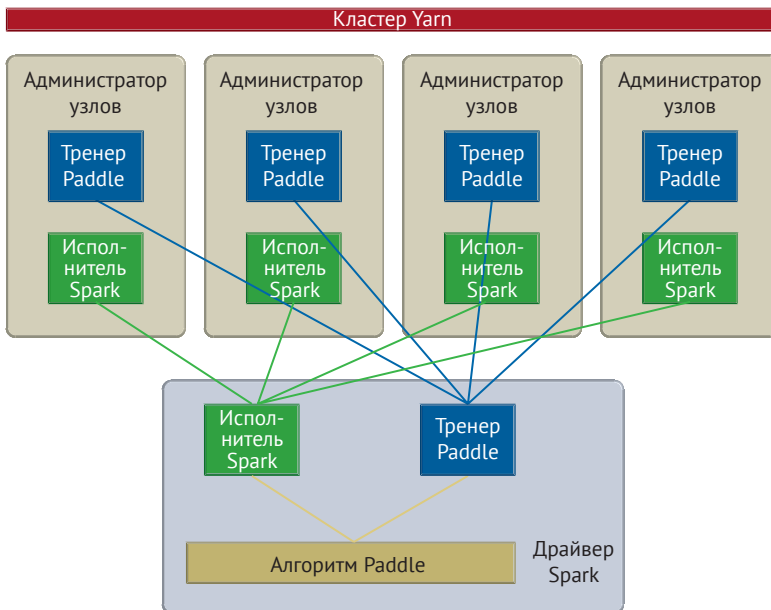


Рис. 1.12. Система обучения распределенной модели глубокого обучения

1.5. Это только начало

Повторимся, что БТС или искусственный интеллект (ИИ) в целом – это не одна технология, а интеграция множества технологий. Поэтому высока необходимость инноваций в алгоритмах, системных интеграциях и облачных платформах. И все это только начало, поскольку в наше время у инженеров есть масса новых возможностей. Вероятно, 2020 год можно считать полноценным началом эры ИИ, поскольку на рынке появилось множество соответствующих инновационных продуктов.

Глава 2

Локализация БТС

Аннотация

Локализация, т. е. точное определение местоположения объекта в режиме реального времени, на сегодняшний день является одной из наиболее важных задач при создании БТС. В этой главе мы рассмотрим различные технические средства локализации, такие как спутниковая навигационная система, лидар и HD-карты, метод визуальной одометрии, а также другие способы счисления координат. Также познакомимся с несколькими реальными примерами применения метода сбора и обобщения данных сенсоров, позволяющего обеспечить более точную локализацию.

2.1. Локализация с помощью спутниковых НАВИГАЦИОННЫХ СИСТЕМ

Когда автомобилем управляет человек, как правило, для определения своего местоположения и построения дальнейшего маршрута, он пользуется *глобальной навигационной спутниковой системой (ГНСС, GNSS – global navigation satellite system)*. Беспилотные автомобили также могут пользоваться GNSS. В этом разделе мы рассмотрим плюсы и минусы применения этой системы в беспилотных технологиях.

2.1.1. GNSS

GNSS представляет собой совокупность нескольких спутниковых систем: GPS, ГЛОНАСС, Galileo и BeiDou. В этом подразделе мы рассмотрим систему GPS. Итак, GPS работает по следующему принципу: приемники GPS улавливают кодированные сигналы, излучаемые спутниками GPS, обрабатывают их и на основе полученных данных вычисляют свое местоположение и скорость в режиме реального времени [1]. Для того чтобы определить свое местоположение, приемнику GPS требуется четыре спутниковых сигнала, которые позволят вычислить положение в трех измерениях и смещение времени. GPS-спутники располагаются в шести орбитальных плоскостях на почти круговых орбитах на высоте около 20 200 км над поверхностью Земли. Наклонение орбит всех спутников составляет 55° к экватору, а период обращения около 11 ч 58 мин.

Сигналы, генерируемые на борту спутников, формируются из опорной (фундаментальной) частоты $f_0 = 10,23$ МГц [1]. Сигналы содержат метку времени бортовых атомных часов, суточная погрешность которых составляет 10^{-13} с. Два несущих сигнала в L-диапазоне частот, обозначаемых как L1 и L2, генерируются путем целочисленного умножения f_0 . Несущие частоты L1 и L2 подвергаются двухфазной модуляции (кодированию) для передачи в приемник сигналов точного бортового времени и параметров орбиты. Коды сигналов представляют собой последовательности с состояниями +1 или -1, соответствующими двоичным значениям 0 или 1. Двухфазная модуляция выполняется сдвигом фазы несущей на 180° всякий раз, когда происходит изменение состояния кода. Спутниковые сигналы содержат информацию о спутниковых орбитах, возмущениях орбиты, системном времени GPS, параметрах ионосферы, а также данные спутниковых часов, сообщения о состоянии системы и т. д. Навигационное сообщение состоит из 25 фреймов, каждый из которых содержит 1500 бит и подразделяется на 5 субфреймов величиной 300 бит.

Еще одной важной частью системы GNSS является определение базовой системы координат, которая имеет решающее значение при описании движения спутника, моделировании наблюдений и интерпретации результатов. GNSS требует наличия двух систем отсчета: (а) инерциальная система отсчета, описывающая движение спутника; и (б) земная система координат, позволяющая описывать местоположение станций наблюдения и результаты спутниковой геодезии. Таким образом, используются две системы. Параметры их преобразования хорошо известны, а сам процесс преобразования происходит в приемнике GNSS посредством программного обеспечения постобработки, что позволяет вычислять положение приемников в соответствии с земной системой координат. Земная система координат строится на трех осях – Z, X и Y. Ось Z совпадает с осью вращения Земли, определенной как *международное условное начало* (CIO – *Conventional International Origin*). Ось X соответствует Гринвичскому меридиану, а ось Y ортогональна осям Z и X и дополняет систему до правой. В качестве системы отсчета GPS используется WGS84, за основу которой взят геоцентрический эквипотенциальный эллипсоид вращения [2].

За последние несколько лет количество GNSS-спутников, поддерживающих несколько спутниковых группировок, росло так же быстро, как и число GNSS-спутников за предыдущее десятилетие. Учитывая появление и развитие новых спутниковых группировок, ожидается, что со временем все устройства будут способны поддерживать несколько группировок, что обеспечит лучшую доступность услуги навигации даже в областях со слабым сигналом; также значительно повысится точность (поскольку чем больше спутников, тем выше точность) и надежность (поскольку сигналы независимых систем труднее подменить).

2.1.1. Анализ погрешностей GNSS

В идеале результаты локализации с применением системы GNSS не должны иметь погрешностей. Однако существуют несколько факторов, ввиду которых погрешности все же имеются. В этом подразделе мы рассмотрим потенциальных виновников их возникновения.

- *Спутниковые часы*: любая погрешность атомных часов спутника GNSS может привести к значительному расхождению между определенными приемником координатами и реальным местоположением объекта. Погрешность часов в 10 нс примерно равна позиционной погрешности в 3 м.
- *Ошибки эфемерид*: спутники GNSS движутся по определенным орбитам. Но параметры орбит, как и параметры спутниковых часов, могут немного меняться. Когда орбита спутника изменяется, наземная система управления отправляет на спутники поправку, и эфемериды спутников обновляются. Однако, даже несмотря на внесенные поправки, погрешности параметров орбиты могут привести к возникновению позиционной погрешности до $\pm 2,5$ м.
- *Ионосферная задержка*: ионосфера – это слой атмосферы Земли на высоте от 80 до 600 км. Ионосфера содержит частицы, имеющие электрический заряд, которые называются ионами. Ионы задерживают спутниковые сигналы и могут быть причиной серьезных погрешностей позиции спутника (обычно ± 5 м). Ионосферная задержка зависит от солнечной активности, времени года, сезона, времени суток и местоположения. Все эти факторы затрудняют прогнозирование влияния ионосферной задержки на спутниковые сигналы. Также данная задержка зависит от радиочастоты сигнала, проходящего через ионосферу.
- *Тропосферная задержка*: тропосфера – это ближайший к поверхности Земли слой атмосферы. Тропосферные задержки зависят от изменения влажности, температуры и атмосферного давления в данном слое. В пределах одной территориальной зоны тропосферные условия практически одинаковы, вследствие чего базовые станции и рабочие приемники испытывают похожее влияние тропосферы. Таким образом, станции *RTK (Real Time Kinematic – кинематика реального времени)* GPS могут компенсировать тропосферную задержку, о чем мы поговорим в следующем подразделе.
- *Многолучевость*: многолучевое распространение возникает, когда сигнал GNSS отражается от какого-либо объекта (например, стены здания) и затем поступает к антенне приемника GNSS. Ввиду того, что отраженный сигнал распространяется дольше, к антенне он поступает с небольшой задержкой. Такая задержка может привести к тому, что координаты, вычисленные приемником, окажутся неверными.

В табл. 2.1 представлены вышеописанные погрешности и их масштаб. Более подробную информацию об этом см. в [3, 4, 5, 6].

Таблица 2.1. Системные погрешности GNSS (на основе [3])

Источник погрешности	Масштаб, м
Бортовые часы	± 2
Ошибка эфемерид	$\pm 2,5$
Ионосферная задержка	± 5
Тропосферная задержка	$\pm 0,5$
Шумы приемника	$\pm 0,3$
Многолучевость	± 1

2.1.3. Спутниковые системы дифференциальной коррекции

Спутниковые системы дифференциальной коррекции (ССДК, SBAS – Satellite Based Augmentation Systems) являются вспомогательными системами GNSS и служат для устранения погрешностей измерений. Они призваны улучшить некоторые характеристики GNSS, такие как точность, целостность, бесперебойность и доступность. Концепция ССДК основывается на слежении за измерениями GNSS с помощью наземных станций, размещенных по всему континенту, координаты расположения которых известны с высокой степенью точности. Погрешности измерений GNSS передаются в пункт обработки, который рассчитывает дифференциальные поправки и параметры целостности. Затем все эти данные транслируются по всему континенту с помощью геостационарных спутников в качестве дополнения к исходному сообщению GNSS или накладываются на него. Геостационарные спутники в свою очередь способны покрывать довольно обширные территории.

Некоторые страны имеют собственные ССДК. В Европе действует *европейская геостационарная служба навигационного покрытия (EGNOS – European Geostationary Navigation Overlay Service)*, покрывающая преимущественно территорию Европы. В США действует *система распространения поправок к данным (WAAS – Wide Area Augmentation System)*. В Китае – *спутниковая навигационная система BeiDou (BDS – the BeiDou System)*. В Японии – *многофункциональная система дифференциальной коррекции спутникового базирования (MSAS – Multi-functional Satellite Augmentation System)*. Индия запустила собственную реализацию ССДК под названием *GPS and GEO Augmented Navigation (GAGAN)*, покрывающую индийский субконтинент. Все вышеперечисленные системы соответствуют единому мировому стандарту, ввиду чего являются совместимыми и способны взаимодействовать между собой.

Обратите внимание, что большинство коммерческих GNSS-приемников поддерживает ССДК. Например, в спецификации системы WAAS прописана точность определения местоположения 7,6 м или меньше как для горизонтальных, так и для вертикальных измерений по крайней мере в 95 % случаев. Фактически показатели измерения эффективности системы показали, что система обеспечивает погрешность не более 1 м по горизонтали и 1,5 м по вертикали на большей части территории США.

2.1.4. Кинематический и дифференциальный режимы GPS

Основываясь на нашем опыте, мы можем утверждать, что большинство существующих на рынке систем GNSS с функцией поддержки нескольких спутниковых группировок обеспечивает точность определения местоположения с погрешностью 2 м. Для водителей-людей такой результат приемлем, а для беспилотного автомобиля – нет, так как транспортному средству необходимо четко знать, где проходит дорога, и видеть ее границы. Чтобы оставаться на определенной полосе, автомобилю необходимо знать, где он находится. Таким образом, требуется локализация хотя бы с дециметровой точностью. В этом индустриям могут помочь режим *кинематики в реальном времени (RTK – Real-Time Kinematic)* и дифференциальный режим GPS. В этом подразделе мы рассмотрим принцип работы этих технологий.

Кинематический режим GNSS обеспечивает высокую точность определения местоположения за счет устранения большего числа погрешностей спутниковых часов, ошибок эфемерид, ионосферных и тропосферных задержек. На рис. 2.1 представлен принцип работы кинематического режима GNSS. Здесь мы можем видеть, что GNSS-приемник установлен на станции, местоположение которой точно известно, т. е. на базовой станции. Приемник базовой станции вычисляет свое местоположение на основе спутниковых данных и сравнивает полученную информацию с координатами его фактического местоположения, а затем определяет разницу. Далее данные с поправками могут быть переданы на борт БА.

RTK работает по принципу измерения фазы несущей и предоставляет более точную информацию о диапазонах (и, следовательно, данные о местоположении), чем кодовые измерения. Локализация с помощью кода – это один из методов обработки, который работает по принципу сбора данных посредством приемника, использующего код «грубого» захвата (*C/A code – coarse acquisition code*). То есть для определения местоположения приемник использует псевдослучайный код. Дифференциальная коррекция позволяет повысить точность вышеописанного метода до 5 м. Принцип измерения фазы несущей также позволяет осуществлять сбор данных через приемник, который для определения местоположения использует сигнал несущей радиочастоты. Частота несущего сигнала гораздо выше, чем частота псевдослучайного кода, поэтому такой способ определения местоположения может предоставить более точные данные, чем определение местоположения с помощью одного лишь псевдослучайного кода. Псевдослучайный код сужает радиус погрешности, но код несущей справляется с этим еще лучше. Дифференциальная коррекция такого метода позволяет повысить точность определения координат до 1 м. При использовании принципа измерения фазы несущей диапазон несущей вычисляется путем определения количества циклов несущей между спутником и БА, а затем умножения этого числа на длину волны несущей. Однако использование этого принципа не исключает появления погрешностей спутниковых часов, ошибок эфемерид, а также ионосферных и тропосферных задержек. Для устранения всех этих проблем с помощью RTK необходимо, чтобы результаты измерений передавались с базовой станции на транспортное средство.

С помощью кинематики в режиме реального времени транспортные средства определяют свое местоположение с помощью таких алгоритмов, как устранение неоднозначности и дифференциальная коррекция. Точность определения местоположения зависит от удаленности транспортного средства от базовой станции и точности дифференциальных поправок. Как правило, точность поправок соответствует точности определения местоположения базовой станции и качеству ее спутниковых наблюдений. Следовательно, критически важным фактором, позволяющим минимизировать воздействия окружающей среды, такие как помехи и многолучевость, является выбор места установки. Также важными факторами являются качество базовой станции, приемников и антенн самих транспортных средств.

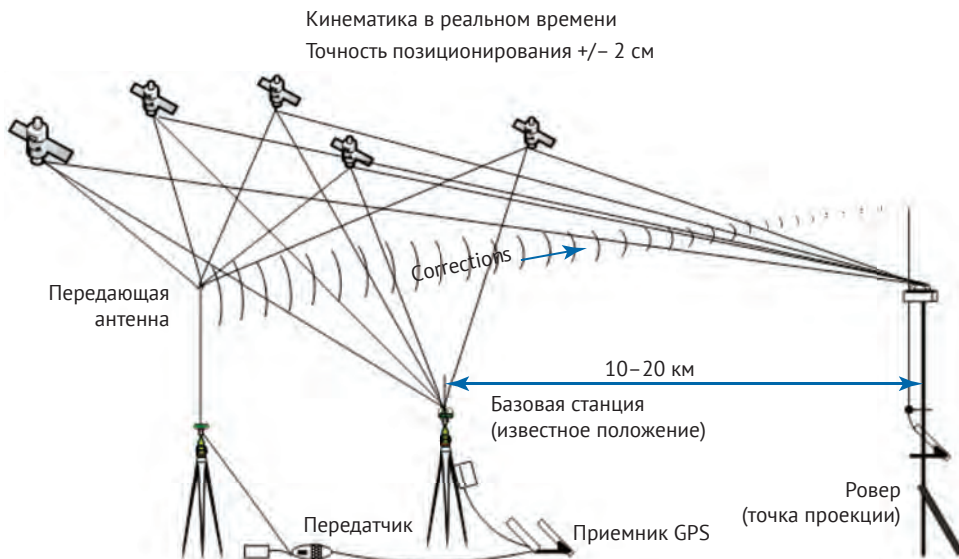


Рис. 2.1. RTK GNSS (на основе [46])

2.1.5. Позиционирование высокой точности

Несмотря на то что система RTK GNSS удовлетворяет требованиям беспилотных технологий, зачастую ее реализация требует развертывания дорогостоящих базовых станций. В этом подразделе мы рассмотрим, как *GNSS-система позиционирования высокой точности (PPP – Precise Point Positioning)* может помочь решить эту проблему [7, 8].

На рис. 2.2 представлен принцип работы PPP GNSS. По всему миру развернуто множество опорных станций, которые получают точные данные об эфемеридах спутников и данные спутниковых часов в режиме реального времени. Затем такие станции вычисляют поправки, которые следует применить к результатам определения местоположения спутника. После расчета поправок они направляются конечным пользователям посредством спутникового сигнала или через интернет. Точно определенные эфемериды спутников и точные данные их часов сводят к минимуму возникновение соответствующих погрешностей. А для устранения ионосферных эффектов первого порядка, пропорциональных частоте несущей волны, можно использовать двухчастотный GNSS-приемник. Следовательно, ионосферная задержка может быть полностью устранена путем использования комбинации двухчастотных измерений GNSS. Тропосферная задержка корректируется путем использования модели UNB (University of New Brunswick – Университет Нью-Брансуика) [9]: для достижения большей точности при оценке местоположения и других неизвестных данных рассчитывается также остаточная тропосферная задержка [10]. Путем сочетания всех вышеописанных методов система PPP может повысить точность определения местоположения до нескольких сантиметров.

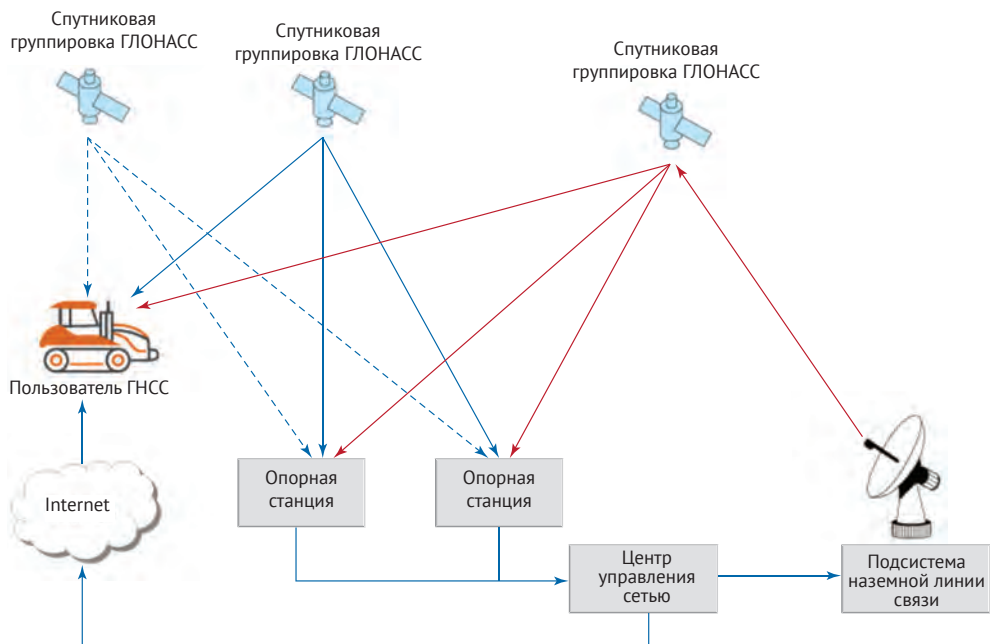


Рис. 2.2. PPP GNSS. Данные предоставлены NovAtel, Inc., использовано с разрешения правообладателя

В частности, алгоритм PPP использует кодовый сигнал и фазовые измерения от двухчастотного приемника, а также точные эфемериды и данные спутниковых часов. Наблюдения, поступающие со всех спутников, обрабатываются с помощью *расширенного фильтра Калмана (EKF – Extended Kalman filter)*. Фильтр Калмана оперирует предположениям значений местоположения, погрешности часов приемника, тропосферной задержки и неоднозначности фазы несущей. Он сводит к минимуму шум в системе и позволяет определять положение с точностью до сантиметра. Оценка состояний совершенствуется с помощью последовательных измерений GNSS до тех пор, пока они не сводятся к стабильным и точным значениям.

Система PPP отличается от RTK тем, что не требует доступа к наблюдениям одной или нескольких базовых станций. Также PPP обеспечивает вычисление абсолютного позиционирования вместо местоположения относительно базовой станции, как это делает RTK. PPP требует лишь точных данных об эфемериде и данных часов, которые вычисляются наземным центром обработки данных с помощью измерений от опорных станций из относительно разреженной сети станций. Обратите внимание, что PPP использует только один приемник GPS и, соответственно, в непосредственной близости от пользователя не требуются опорные станции. Следовательно, PPP можно рассматривать как систему глобального позиционирования, потому что его принцип расчета координат относится к глобальной системе отсчета. PPP обеспечивает гораздо большую согласованность позиционирования, чем RTK, где решения принимаются относительно локальной базовой станции

или станций. Кроме того, по своей структуре система PPP аналогична системе дифференциальной коррекции. В сравнении с этой системой, ключевым преимуществом PPP является то, что ей достаточно наличия точных эталонных эфемерид и данных спутниковых часов в режиме реального времени для обеспечения сантиметровой точности, в то время как погрешность системы дифференциальной коррекции может составлять несколько метров. Кроме того, системы PPP позволяют использовать единый глобальный поток поправок, в то время как системы дифференциальной коррекции работают на уровне регионов.

Одна из основных проблем PPP заключается в том, что для устранения локальных помех, таких как атмосферные условия, многолучевость и геометрия спутника, и обеспечения достаточной точности определения координат (с погрешностью в дециметр) обычно требуется длительный период времени (около 30 мин). На сегодняшний день существует несколько консолидированных сервисов постобработки систем PPP. Но в то же время такие системы находятся на начальной стадии разработки [11, 12, 13].

2.1.6. Интеграция GNSS и INS

В предыдущих подразделах мы рассмотрели различные поколения технологий GNSS. В этом подразделе мы исследуем, как инерционные данные могут быть использованы для улучшения методов локализации GNSS [14]. *Инерциальная навигационная система (INS, INS – Inertial Navigation System)* использует информацию о вращении и ускорении от *инерциального измерительного модуля (IMU – Inertial Measurement Unit)* для вычисления относительного положения объекта с течением времени. Типичный шестиосевой модуль состоит из шести дополнительных датчиков, размещенных на трех ортогональных осях. На каждой оси расположены акселерометр и гироскоп. Акселерометры измеряют линейное ускорение, а гироскопы измеряют ускорение вращения (угловое ускорение). С помощью этих датчиков инерциальные измерительные модули могут точно определять свое относительное движение в трехмерном пространстве. INS использует эти измерения для расчета местоположения и скорости. Кроме того, модули способны измерять скорости вращения относительно трех осей. Впоследствии эти измерения могут быть использованы для определения относительных положений (крен, тангаж и азимут).

Как правило, системы INS работают с частотой 1 кГц, что обеспечивает частое обновление данных о местоположении. Однако такие системы имеют и недостатки: во-первых, INS обеспечивает только определение положения относительно начальной точки, которая также должна быть известна INS. Во-вторых, что более важно, навигация в трехмерном пространстве с помощью INS фактически представляет собой суммирование (или интеграцию) сотен/тысяч выборок в секунду, во время создания которых также возникают и накапливаются погрешности. Это означает, что без коррекции и наличия внешнего опорного сигнала система INS будет быстро отклоняться от истинного положения. Таким образом, при использовании INS для выполнения задач локализации крайне важно обеспечить точный внешний ориентир, позволяющий минимизировать погрешности локализации с помощью математического фильтра – фильтра Калмана.

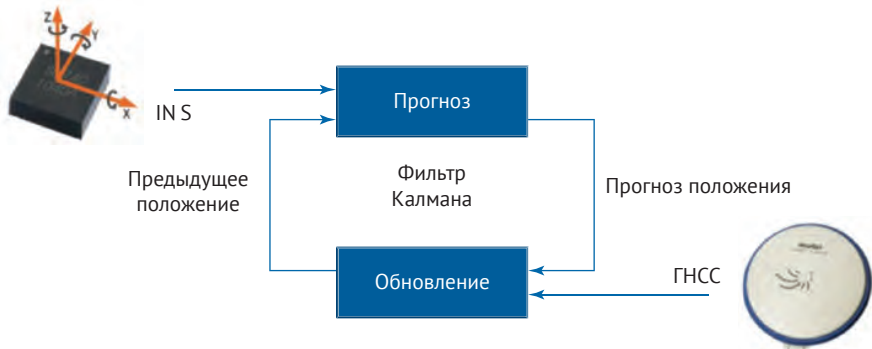


Рис. 2.3. Интеграция GNSS и INS с фильтром Калмана

Как представлено на рис. 2.3, внешним ориентиром может являться система GNSS. Спутниковая система предоставляет абсолютный набор координат, который можно использовать в качестве начальной точки. Также она предоставляет данные о координатах и скорости непрерывно, что позволяет обновлять оценки фильтра INS. Когда сигнал GNSS прерывается (например, при проезде транспортного средства через туннель), система INS поможет определять местоположение транспортного средства в течение короткого периода времени.

2.2. Локализация с помощью лидара и HD-карт

Большинство производителей прототипов БТС, в том числе Waymo, Baidu, BMW и др., для решения задач локализации в своих проектах используют лидары и HD-карты. В этом разделе мы рассмотрим, как устроены их технологии, а также изучим аспекты их взаимодействия, позволяющие обеспечить точную локализацию БТС.

2.2.1. Технология LiDAR

В этом подразделе мы рассмотрим технологию LiDAR более подробно. Название устройства «лидар» произошло от английского сокращения LiDAR, которое расшифровывается как *Light Detection And Ranging* и переводится на русский язык как «обнаружение и определение дальности с помощью света». Сенсор лидара измеряет расстояние до объекта путем освещения его импульсным лазерным излучением и дальнейшего измерения времени, через которое отраженный от объекта импульс возвращается к источнику [15]. Затем на основе разницы в значениях времени возврата лазерного импульса и длины волны создается цифровая 3D-модель объекта. Таким образом, основной принцип работы лидара заключается в следующем: сначала излучатель испускает лазерные импульсы на поверхность объекта (частота импульсов может достигать 150 000 Гц). Затем лидар измеряет время, затраченное на прохождение каждого импульса. И, поскольку скорость света является постоянной величиной, прибор может рассчитать расстояние между излучателем и объектом с высокой точностью. Путем повторения этого процесса снова и снова прибор может построить сложную «карту» измеряемой поверхности.

Как правило, лидары используют один из двух методов обнаружения объектов: некогерентное (также известное как прямой метод измерения) и когерентное обнаружение [16]. Когерентные системы лучше всего подходят для доплеровских или фазочувствительных измерений. Обычно такие системы принимают сигналы методом оптического гетеродинирования, который представляет собой способ извлечения информации, закодированной в виде модуляции фазы или частоты электромагнитного излучения. Благодаря этому мощностные затраты таких систем значительно снижаются, отчасти за счет более высоких требований к трансиверу. Для сравнения: когерентный свет распространяется во всех направлениях, а в некогерентных системах применяются узкоспециализированные диоды, которые генерируют энергию в оптической части электромагнитного спектра или около нее, что означает, что все электромагнитные волны распространяются в одном направлении. Таким образом, потребление энергии значительно снижается.

Существуют две основные категории импульсных лидаров: *высокоэнергетические* и *микроимпульсные* системы. Высокоэнергетические лидары испускают более мощные световые импульсы, которые могут быть опасны для глаз человека. Обычно такие системы используются для исследования атмосферы, в частности для измерения различных атмосферных параметров, таких как высота, наслоения и плотность облаков, свойства частиц облаков, температура, давление, ветер, влажность и концентрация примесей газов. Микроимпульсные системы, напротив, излучают свет меньшей мощности и являются безопасными для человеческих глаз. Но даже при их использовании требуется соблюдать небольшие меры предосторожности. Оригинальная конструкция [17] микроимпульсных лидар-передатчиков представляет собой диодный лазер с энергией импульса в микроджоулях и высокой частотой их испускания. За счет расширения испускаемого луча достигается безопасность для человеческого глаза. В приемнике размещается твердотельный лавинный фотодиодный детектор, работающий по принципу счетчика Гейгера. При создании БТС обычно применяются когерентные микроимпульсные лидар-системы, поскольку они соответствуют первой категории требований лазерной безопасности.

Лазеры, используемые в лидарах, можно разделить на категории согласно длине волны. Чаще всего используются лазеры с длиной волны 600–1000 нм, и обычно их максимальная мощность ограничивается в соответствии с требованиями первой категории лазерной безопасности. Также достаточно часто используются лазеры с длиной волны 1550 нм. Такие лазеры позволяют обнаруживать объекты на большей дальности, однако с меньшей точностью. Кроме того, лазер с длиной волны 1550 нм не виден для приборов ночного видения и поэтому хорошо подходит для использования в военных целях. Лидар-системы, установленные на бортах самолетов, оснащены YAG-лазерами с диодной накачкой 1064 нм, а в батиметрических системах (измерение глубины водоемов) используются YAG-лазеры с двойной диодной накачкой 532 нм, излучение которых меньше подвержено затуханию в воде. Более четкого разрешения можно достичь путем излучения коротких импульсов при условии, что полосы пропускания детектора приемника и электроники достаточно, чтобы справиться с увеличившимся потоком данных.

Как правило, лидар-системы состоят из двух основных групп компонентов: лазерных сканеров и лазерных приемников. Скорость генерации изображений зависит от скорости, с которой они могут быть отсканированы системой. На сегодняшний день различным целям соответствуют различные методы сканирования, среди них колебание излучателя по азимуту и углу места, системы колеблющихся плоских зеркал, двухосевые сканеры и многогранные зеркала. Тип оптики определяет разрешение и дальность обнаружения системой [18, 19]. Лазерные приемники считывают и записывают сигнал, принимаемый системой. Существует два основных типа технологий лазерных приемников: кремниевые лавинные фотодиоды и фотоумножители [20].

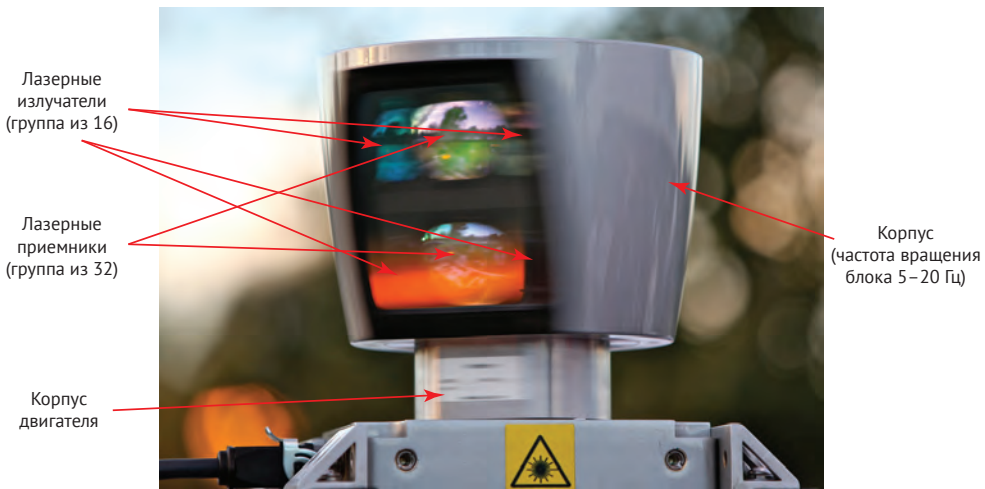


Рис. 2.4. Velodyne HDL-64 [45]

На рис. 2.4 представлена лидар-система Velodyne HDL-64, которая широко применяется при создании БТС. Данная система использует 64 лидар-канала, позиционированных по углам от $+2,0^\circ$ до $-24,9^\circ$ для формирования вертикального углового поля зрения $26,9^\circ$ и обеспечивающих горизонтальный обзор 360° в реальном времени благодаря конструкции с вращающейся головкой. Частота вращения головки устанавливается пользователем в диапазоне 5–20 Гц, что позволяет определять плотность точек данных, генерируемых лидаром. Лазер устройства HDL-64 имеет длину волны 905 нм и импульс 5 нс. Он способен захватывать облако точек с частотой до 2 200 000 точек/с на расстоянии до 120 м и с точностью ± 2 см. Верхняя часть устройства состоит из лазерных излучателей (4 группы по 16 штук в каждой), а нижняя часть состоит из лазерных приемников (2 группы по 32 штуки в каждой).

На практике одной из основных проблем, с которыми сталкиваются производители и пользователи лидаров, является калибровка [21]. Точность лидаров сильно зависит от их калибровки. При должной калибровке точные трехмерные данные из окружающей среды могут быть легко обработаны для выделения линейных или плоских объектов (очертаний). И напротив, выделение объектов затрудняется, если сенсор плохо откалиброван.

Многолучевой лидар можно представить в виде набора лучей (прямых линий). Они определяют положение и ориентацию лазерных лучей в привязанной к сенсору системе координат. Внутренняя калибровка для таких систем заключается в оценке параметров, определяющих положение и ориентацию каждого из лазерных лучей. Принцип, лежащий в основе методов калибровки, – это процесс оптимизации, выполняемый для оценки параметров калибровки лидара таким образом, чтобы 3D-данные, полученные лидаром, максимально соответствовали действительности. Процесс калибровки представляет собой процесс оптимизации, который включает в себя множество параметров и может быть разделен на следующие этапы:

- *выбор параметризации*: для определения одного лазерного луча в 3D-системе координат требуется не менее пяти параметров, включая два параметра угла для определения положения линии луча в пространстве и три параметра для определения начальной точки луча. Если для корректировки измерения, выполненного лазерным лучом, требуется коэффициент коррекции расстояния, количество параметров калибровки увеличивается до шести или семи на каждый лазерный луч;
- *выбор целевой функции*: основой процесса оптимизации является выбор целевой функции / функции стоимости (переменная C). Во время этого процесса осуществляется количественное сравнение полученных данных трехмерного облака точек и реальной среды. Согласно формуле функции стоимости, если наблюдается большая разница между полученными данными и реальной средой, затраты будут выше, и, наоборот, при соответствии между полученными трехмерными данными и реальной средой затраты будут ниже;
- *сегментацию данных*: этот шаг предназначен для фильтрации данных, которые фактически соответствуют объекту калибровки, о котором которому известны достоверные данные. Среда, выбранная для процесса калибровки, должна быть спроектирована и сделана так, чтобы обеспечить соответствующую сегментацию данных.

Количество калибровочных параметров увеличивается пропорционально количеству лучей. Следовательно, процесс калибровки для устройств с большим количеством лучей будет сложнее. Это одна из основных причин, почему стоимость устройств с многолучевыми системами гораздо выше.

2.2.2. HD-карты. Краткий обзор

В этом подразделе мы рассмотрим технические детали создания HD-карт. Но сначала ответим на вопрос: «Зачем нужны HD-карты для беспилотной технологии?» Представьте на мгновение поездку по знакомому маршруту, например, из дома в офис. Еще до того, как вы начнете путь, в вашей голове уже будет мысленная «карта» этой поездки, что позволит вам сосредоточиться на действительно важных для безопасности аспектах вождения. Например, вы можете предвидеть необычное поведение других водителей, которые решат объехать знакомую вам яму. Также вы можете заранее вспомнить ограничение скорости, даже не смотря на знаки, которые вдруг закроет большой грузовик. Теперь сравните это с поездкой по совершенно новому маршруту. Здесь вашему мозгу нужно будет обработать гораздо больше информации, потому что

окружающая среда ему незнакома. Вы можете реагировать только на то, что видите в данный момент. Те же принципы применимы к БТС. HD-карты делают маршруты знакомыми для БТС, что обеспечивает безопасность в поездках. Тогда почему бы не использовать для беспилотных технологий цифровые карты? Ответ прост. Цифровые карты предназначены для использования людьми. Обычно они имеют низкое разрешение (точность на уровне метра) низкую частоту обновления. Но для того, чтобы БТС двигалось по дороге, ему необходимо знать, где находится эта дорога. Чтобы оставаться на определенной полосе, ему необходимо знать, где она находится. Таким образом, локализация должна осуществляться с точностью до дециметров. Поэтому цифровые карты не смогут обеспечить БТС точную локализацию в режиме реального времени, особенно когда окружающая среда не соответствует тому, что указано на карте.

При создании карт для применения их в беспилотных технологиях возникают три основные проблемы: карты должны быть очень точными (с точностью до сантиметра), соответственно, в HD-качестве; они должны постоянно обновляться, чтобы отражать изменения на дорогах (на практике частота обновления составляет один раз в неделю); они должны бесперебойно и с высокой производительностью взаимодействовать с остальными технологиями БТС. Для достижения высокой точности мы можем использовать лидар в сочетании с другими сенсорами для сбора картографических данных. Для того чтобы обеспечить высокую частоту обновления, можем использовать краудсорсинг (источник информации – люди) во время процесса создания карты (подход DeerMap) вместо того, чтобы периодически создавать карты путем использования специальных сканирующих автомобилей (подход Google и Baidu). Для интеграции HD-карт с беспилотными технологиями мы можем создать высокопроизводительную облачную инфраструктуру, которая будет предоставлять обновления HD-карт в реальном времени. Далее в этом разделе мы рассмотрим создание HD-карт, погрешность которых составляет лишь несколько сантиметров.

Основная цель создания HD-карт состоит в том, чтобы улучшить навигацию GNSS/INS за счет использования подробной карты окружающей среды, а затем использовать лидар для определения местоположения транспортного средства относительно HD-карты [22, 23, 24]. Таким образом, ключом к созданию HD-карт является объединение информации, полученной различными способами (GNSS/INS/лидар) Такой подход позволяет минимизировать погрешности в каждом элементе сетки HD-карты. Сначала GNSS/INS генерируют приблизительную информацию о местоположении для каждого сканирования, а затем лидар определяет местоположение каждой точки с высокой точностью. Ключевая проблема здесь заключается в том, каким образом обеспечить совместную работу различных источников данных для получения высокоточных локальных карт и как создать глобальные карты путем «сшивания» локальных карт вместе.

Для начала давайте изучим анатомию HD-карт. Подобно традиционным картам, HD-карты также имеют иерархические структуры данных. Базовый слой (нижний уровень) представляет собой высокоточный 2D-объект с разрешением 5×5 см [23]. Этот уровень включает 2D-вид сверху поверхности дороги, снятый в инфракрасном спектре с помощью лидара. Каждый элемент сетки на этом слое хранит информацию о способности отражать импульсы лидара.

По информации об отражательной способности поверхности мы можем выяснить, является ли элемент сетки свободной дорожной поверхностью или на этом участке есть препятствия. Как будет сказано несколько позже, для локализации БТС с помощью HD-карт в режиме реального времени необходимо сравнить текущие лидар-сканы с информацией об отражательной способности, которая хранится в элементах сетки базового слоя.

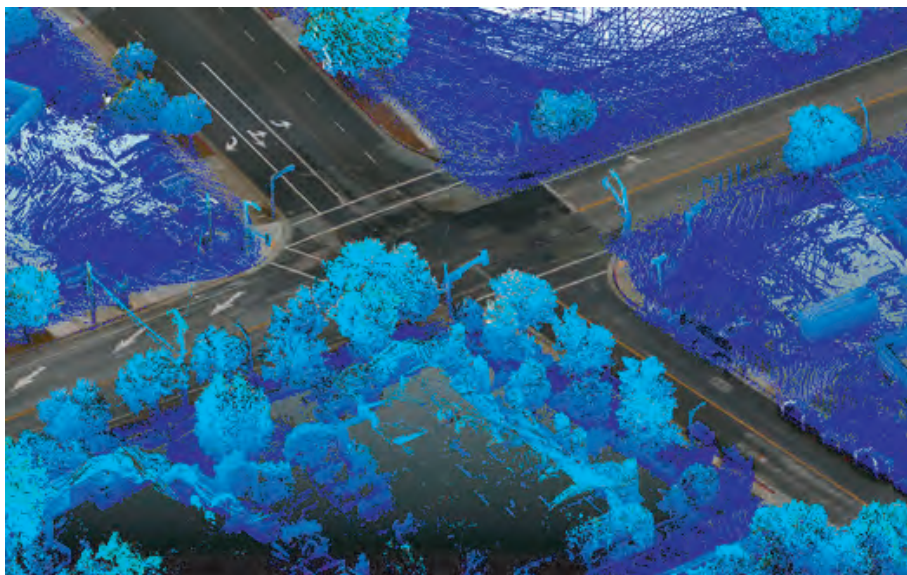


Рис. 2.5. HD-карта. Любезно предоставлено ДеерМар, используется с разрешения правообладателя

Базовый слой представляет собой ортогональный инфракрасный снимок Земли. Каждой ячейке 2D-сетки назначается положение x - y в среде со значением отражательной способности инфракрасного излучения. Для захвата необработанного изображения на транспортном средстве устанавливаются один или несколько лидаров, направленных вниз на поверхность дороги. Лазеры в этих сенсорах не только определяют расстояние до поверхности, но и измеряют коэффициент отражательной способности. Путем наложения данных отражательной способности на данные 3D-измерений в результате мы получаем инфракрасное изображение поверхности. Одним из стандартных способов, позволяющих исключить влияние нестационарных объектов на результаты локализации, является подгонка плоскости земной поверхности к каждому результату лазерного сканирования и сохранение только тех измерений, которые совпадают с этой плоскостью [23]. Ключевым преимуществом лидара, по сравнению с обычными камерами, является возможность игнорировать вертикальные объекты. Таким образом, на слое карты отображается только плоская поверхность, а другие транспортные средства автоматически исключаются. Такая система может работать даже ночью, поскольку лидар не нуждается в освещении. Следовательно, процесс создания карт практически не зависит от условий внешнего освещения (в отличие от пассивных камер).

Каждый скан, полученный лидарами, фактически является локальной картой окружающей среды. Однако для создания глобальной карты необходимо «сшить» все локальные карты. Этот процесс называется *сопоставлением карт* [24]. Сущность процесса сопоставления карт заключается в сравнении локальных сканов лидаров и определении мест, где они совпадают друг с другом. Далее, эти места используются как ориентиры для «сшивания» карт. Так, мы можем представить две последовательности сканов как две непересекающиеся последовательности временных показателей: a_1, a_2 и т. д. и b_1, b_2 и т. д. Соответствующие друг другу ячейки сетки отражают наложение сканов друг на друга, превышающее заданный порог T . Как только область пересечения сканов будет найдена, можно сгенерировать две отдельные карты: одну на основе данных a_1, a_2 и т. д., а другую на основе b_1, b_2 , и т. д. Затем выполняется процесс совмещения карт. Во время этого процесса предполагается, что поиск пересечения выполняется в соответствии с максимальной вероятностью отражения инфракрасного излучения. В частности, между этими картами вычисляется поле линейной корреляции различных смещений x – y . Обратите внимание, что все сканы лидара подвергаются постобработке в соответствии с данными GNSS и INS. Таким образом, каждый скан связан с исходным положением $\langle x, y, \theta \rangle$, где x и y – точное местоположение транспортного средства, а θ – направление его движения. Данные о положении позволяют минимизировать погрешности в процессе сопоставления карт, во время которого вычисляются коэффициенты корреляции для элементов, значение инфракрасной отражательной способности которых уже известно. В случаях, когда участок пересечения карт является уникальным, мы находим единственный пик в этом поле корреляции. Тогда предполагается, что пик этого корреляционного поля является наилучшей оценкой для локального совмещения карт.

Следующим слоем HD-карты после базового является слой семантической информации. Как представлено на рис. 2.6, этот слой содержит информацию о местоположении, а также характеристиках линий дорожной разметки и полос движения. В различных условиях (например, при плохой погоде или наличии помех от других транспортных средств) надежность информации, поступающей с сенсоров БТС, может снижаться. В таких случаях информация о полосах движения на HD-карте может помочь БТС точно и надежно определять положение дорожных полос, а также оценивать степень загруженности соседних полос в режиме реального времени. Также в HD-картах есть слой, отображающий информацию о дорожных знаках, сигналах светофоров и т. п. Этот слой несет в себе две функции: во-первых, на этапе восприятия он позволяет автомобилю «увидеть» дорожные знаки и ограничения скорости; во-вторых, на этапе планирования он позволяет транспортному средству передвигаться безопасно, ориентируясь на дорожные знаки и ограничения скорости, содержащиеся на HD-карте, даже в тех случаях, когда оно не может обнаружить их в реальной среде.

Еще одна проблема заключается в ограниченном запасе памяти, ведь, как известно, HD-снимкам требуется много места для хранения. Например, значительный объем памяти занимают карты обширных пространств с разрешением 5 см/пиксель. Для сокращения занимаемого картами объема существует два метода, позволяющих помещать эти данные в основную память [23]. Первый метод заключается в отфильтровывании нерелевантной информации:

при сборе данных в движущемся транспортном средстве прямоугольная область, ограничивающая размер скана, увеличивается квадратично с пройденным расстоянием, несмотря на то что сами данные растут только линейно. Для того чтобы этого избежать, мы можем разбить прямоугольную область на квадратную сетку и сохранять только необходимые ячейки. Такой подход позволяет снизить объем памяти, занимаемой изображением до 10 Мб на милю дороги при разрешении 5 см/пиксель. Таким образом, на жестком диске емкостью 1 Тб может храниться информация об окружающей среде на расстояние 100 000 миль. Итак, первый метод оптимизирует использование дискового накопителя. Второй же метод обеспечивает оптимизацию использования памяти иным способом. В любой момент во время движения нам требуется лишь локальная HD-карта. Кроме того, у нас имеются данные GNSS/INS, которые помогают приблизительно определять положение транспортного средства в реальном времени. Следовательно, мы можем использовать эту информацию для динамической предварительной загрузки небольшой части HD-карты в память независимо от размера всей карты.

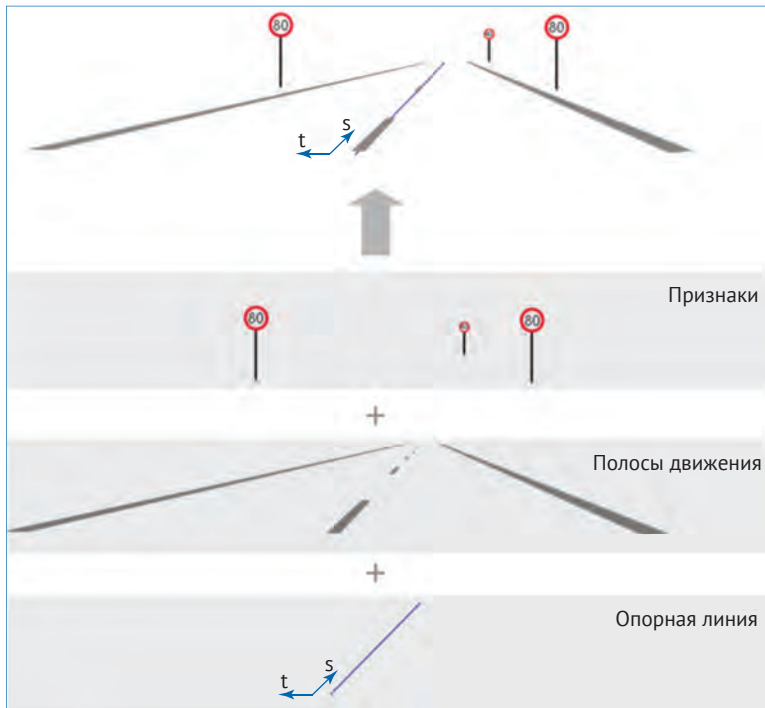


Рис. 2.6. Семантический слой HD-карт