

**Санкт-Петербургский Государственный Университет
Математико-Механический факультет**

Подсчет расстояния с помощью GPS координат

**Курсовая работа студентки 344 группы
Лях Юлии Михайловны**

Научный руководитель:

А. М. Боташев

**Санкт-Петербург
2015 год**

Оглавление:

Введение	3
Постановка задачи	4
Обзор литературы и существующих решений.....	4
Реализация.....	7
Анализ ошибок GPS.....	7
Сглаживание шумов - фильтр Калмана.....	9
Метод геоквадратов.....	11
Отсев прыжков.....	13
Тестирование и результаты.....	16
Список литературы.....	20

Введение

На сегодняшний день службы геолокации широко используются в мобильных приложениях — GPS данные привязываются к фотографиям, используются в картах, для планирования маршрутов в приложениях-навигаторах. Несмотря на заявленную погрешность примерно в 13 метров, на деле показания различных приемников сильно разнятся. Например, точность падает среди высоких зданий, деревьев, в плохую погоду и при других обстоятельствах. Для получения одной примерной точки этой точности достаточно. Однако, если использовать поток геоданных для вычисления расстояния, то ошибки накапливаются, и результат получается значительно хуже ожидаемого. Ситуация с вычислением расстояния на бюджетных смартфонах оказалась критической — погрешность при вычислении расстояния может превышать 200-300%. Примеры таких ошибок встретились в ходе анализа информации о поездках, когда дистанция вычислялась последовательно путем прибавления расстояния между двумя точками.

Обычно для уточнения координат пути используется векторная карта дорог города. Тем самым мы предполагаем, что автомобиль движется по дороге и проектируем исходную точку на ближайший вектор. По такому принципу осуществляют навигацию большинство популярных сервисов карт, например Google Maps и Яндекс карты. Однако, это возможно применить лишь в том случае, когда уже существует актуальная и надежная база координат дорог. Для того чтобы построить ее необходимо либо вручную картографировать все дороги, либо объехать весь город на машине с GPS-трекером, чьи показания можно считать корректными. Все это требует больших ресурсов, как материальных, так и человеко-временных. Таким образом возникает задача максимально улучшить погрешность доступных GPS приборов лишь программно, без дополнительных ресурсов, а так же без знаний о самих приборах.

Постановка задачи

Таким образом, в задачи данной курсовой работы входит разработка алгоритма, который будет минимизировать ошибки при подсчете расстояния с помощью GPS данных. Допускается погрешность, большая заявленной GPS системой, но требуется устойчивость ошибки — на одних и тех же маршрутах расстояние не должно сильно различаться. Кроме того, алгоритм должен производить обработку данных динамически и в реальном времени, или приближенно к нему на несколько секунд, для того чтобы пользователь мог отслеживать текущее расстояние в поездке. Алгоритм должен обрабатывать кратковременную потерю сигнала, шумы и прочие ошибки приемника, которые появятся в ходе анализа и тестирования. Так как предполагается, что расчет расстояния будет происходить на мобильном устройстве, сложность вычислений должна быть небольшой, чтобы любой смартфон справлялся с задачей без задержек. Более того, необходима возможность использовать алгоритм сразу же, без необходимости собирать и анализировать какую-либо информацию о городе.

Обзор литературы и существующих решений

Многие приложения-карты осуществляют уточнение координат. Например, если мы пользуемся Google Maps, то видим свое перемещение на карте довольно точно, но если посмотреть на сырые координаты GPS устройства, то видно, что множество разбросанных точек проектируется на дороги. Программно можно использовать возможности Google Maps с помощью нескольких интерфейсов: Roads, Directions и Geolocation. Больше всего подходит Google Maps Roads API [4], с его помощью можно получить информацию, по какой дороге едет автомобиль и некоторые ее свойства, например разрешенная скорость. Однако, свободное использование этого инструмента ограничено всего

лишь 2500 запросами в день. При запросе каждую секунду, этого хватает всего на 40 минут поездки, чего слишком мало для практического использования.

Существует другая интересная идея для получения местоположения - она реализована в интерфейсе Google Maps Geolocation API а так же в приложении Яндекс Локатор [5]. Оба этих продукта предоставляют примерное местоположение пользователя без использования GPS координат, основываясь на данных о ближайших GSM вышках, wi-fi роутерах, а так же ip адресе устройства. Такой метод возможен на устройствах, которые могут получить информацию о силе различных сигналов, в том числе, смартфоны с операционной системой Android. Однако любая реализация похожего алгоритма так же требует сбора и анализа большого количества информации о местоположении источников сигнала. Яндекс собирал данные с пользователей Яндекс карт, однако проблему больших ошибок требовалось решить быстро, а использование сторонних продуктов так же накладывало серьезные ограничения на количество запросов в день. Однако обе идеи могут существенно повысить точность подсчета расстояния, так что можно анализировать и применять их в будущем, после устранения критических ошибок.

Одно из существующих решений задачи фильтрации координат, представленных в статье — это сбор и первичная фильтрация координат маршрутов, которые мы после этого считаем корректными и используем для проекции новых маршрутов [1]. О причинах, делающих невозможным сбор информации, было уже сказано, однако идея фильтрации подходит и для реализации нашего алгоритма. Основной идеей здесь служит объединение соседних точек в некоем радиусе в одну. Это позволяет видеть немного более усредненную картину процесса. В статье [2] такой подход применяется в качестве оптимизации - и точки в квадрате 100 на 100 метров объединяются, а для быстрого поиска ближайших вершин используется k-мерное дерево.

Распространенным методом устранения разнообразных шумов и погрешностей в показаниях датчиков являются математические

фильтры - такие как фильтр Калмана [3] и среднее скользящее. Различные реализации действительно хорошо показывают себя на GPS координатах, тем более, что он основывается лишь на текущем и предыдущем значении датчиков, а значит, можно использовать его в реальном времени. Однако, для его реализации необходимо знать дисперсию ошибки у датчика, а в каждом конкретном случае она неизвестна. Его можно использовать в алгоритме фильтрации, при том условии, что удастся подобрать оптимальное значение параметров, чтобы сглаживались шумы, но не сглаживались правильные и довольно резкие повороты на перекрестках.

Реализация

Анализ ошибок GPS

Для того чтобы провести анализ возникающих ошибок при подсчете расстояния, было реализовано мобильное приложение-клиент.

В первой реализации приложение работало следующим образом - весь поток координат мобильное устройство передавало серверу, где и происходил последовательный подсчет расстояния.

Подсчитанное значение возвращалось клиенту и показывалось пользователю. Было решено производить фильтрацию и подсчет дистанции на устройстве-клиенте и отправлять на сервер только реальное расстояние, тем самым снизив нагрузку на передачу данных.

Перед тем как реализовать фильтрацию данных, было решено проанализировать и выделить те ошибки, которые возникают в ходе подсчета расстояния. Для этого в течение поездок собирались отчеты со всеми координатами, на них позже тестировалась работа различных фильтров. Было выделено несколько типов повторяющихся ошибок, которые больше всего мешают корректно посчитать дистанцию.

Первый тип ошибок, от которых не застрахованы даже дорогие GPS-трекеры, это шумы (см. рис. 1). С такими ошибками часто сталкиваются автомобилисты, если трекер находится в постоянно во включенном состоянии - например на парковке, когда автомобиль просто стоит на месте, GPS координаты все равно колеблются в некоем радиусе. При соединении получается “звезда”, и расстояние будет ненулевым. В движении это выглядит как ломаные зигзагообразные траектории на прямых участках дороги.



рисунок 1. Иллюстрация шумов на карте Google

Другая распространенная ситуация на слабых смартфонах - это потеря связи и сигнала со спутника. Восстановить недостающую информацию, конечно, сложно, но необходимо хотя бы минимизировать возможные последствия таких ошибок.

Третий тип ошибок, который удалось выделить - сопряжен в некотором смысле с предыдущим. На некоторых телефонах при потере связи со спутников возникала интересная ситуация “прыжков” - GPS приемник говорил о своем местоположении в одной какой-то точке, обычно весьма удаленной от реального маршрута (см. рис. 2). Возникло предположение, что в таких точках находится GSM вышка или подобное устройство, потому что за поездку может набираться до нескольких сотен таких скачков, что критично влияло на результирующую дистанцию, однако точно выяснить причину этих ошибок не удалось. Таким образом, преимущественно ошибочная дистанция оказывается значительно больше настоящей.

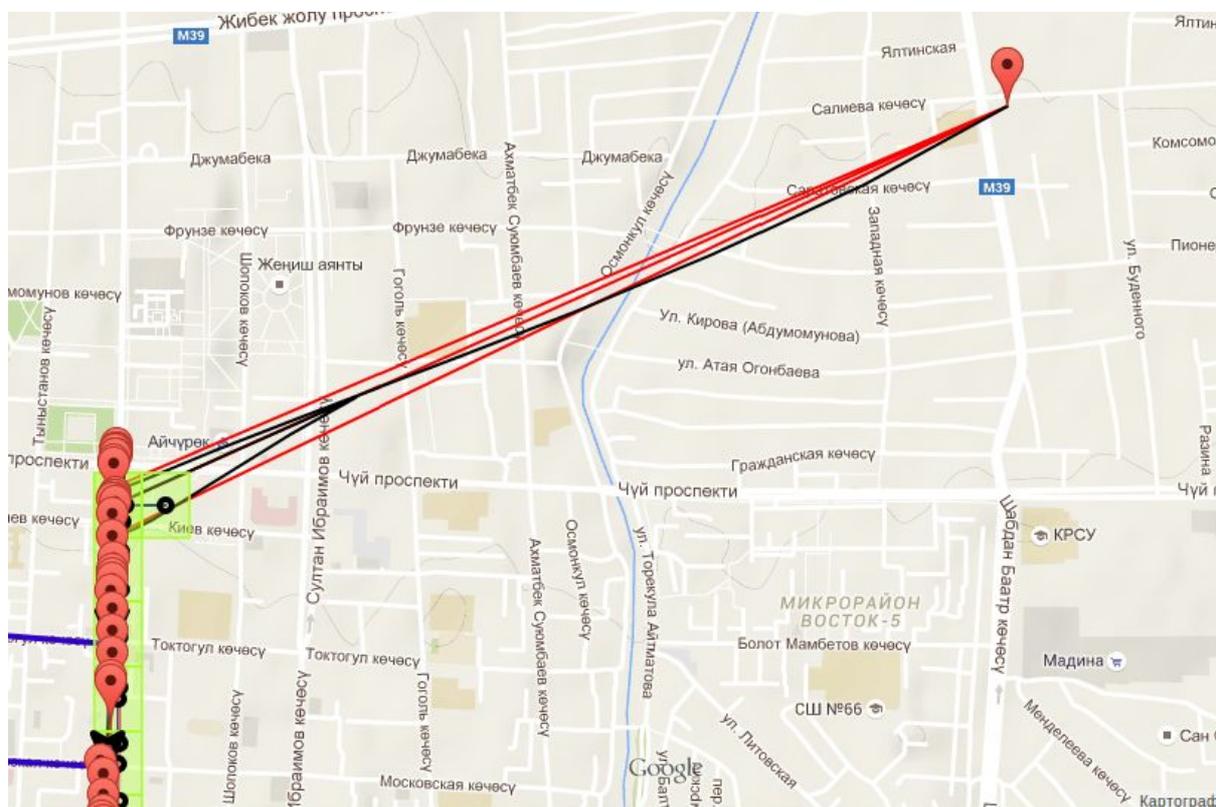


рисунок 2. Иллюстрация “прыжка”

Сглаживание шумов

Для того чтобы отсеять некоторые зигзагообразные шумы был реализован фильтр Калмана (пример использования см. на рис. 3). Выбор обусловлен тем, что это довольно мощный математический аппарат, который способен обрабатывать данные на лету, не накапливая их для анализа. Результатом работы фильтра является сглаженная траектория движения.



рисунок 3. Пример использования фильтра Калмана для сглаживания потока GPS координат.

Сложность его использования заключается в том, что для использования необходимо знать значение среднеквадратичного отклонения ошибки, что в общем случае узнать невозможно для универсального алгоритма. Однако изменяя этот параметр произвольным образом, можно получить разной силы сглаживание. Здесь сложность возникает в выборе этого параметра - можно сильно сгладить траекторию и получить верную ровную линию, но в ситуации с частыми поворотами, острые углы перекрестков так же станут круглее и нужное расстояние потеряется (см. рис. 4). И конечно, этот фильтр не способен отсеять прыжки. Так что их предстояло отсеять на каком-то предварительном этапе. Было решено использовать сглаживание с малым значением параметра - это позволило немного снизить шумы, не нарушая геометрию дорог. Полученный результат в большинстве случаев далек от реальной картины поездки, но хорошо подходит для следующего фильтра.

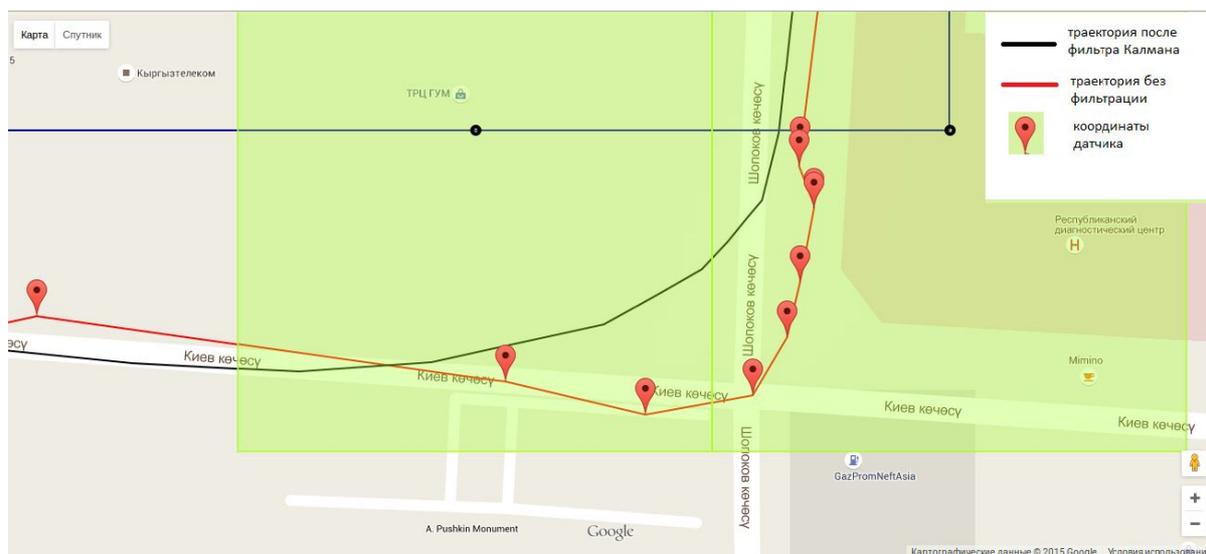


рисунок 4. Иллюстрация излишнего сглаживания при использовании фильтра Калмана.

Метод геоквадратов

Следующим фильтром был выбран отсев близких точек. Это довольно распространенная идея - взять несколько точек в определенном радиусе и объединить их в одну. Идея хороша тем, что позволяет так же видеть более усредненную траекторию, и отсеять множество избыточной информации, тем самым снизив поток данных через сеть.

Вопросы возникают в том, какой радиус брать, и какую точку брать в качестве средней. Другая сложность состоит в том, что вычисление расстояния в сферических координатах - довольно сложная процедура. Выполнять эти вычисления для приходящей каждую секунду новой точке было бы слишком расточительной тратой ресурсов мобильного телефона. Так что возникла необходимость быстро находить точки в одном радиусе. Для этого была реализована гео-хэш функция, которая по широте и долготе возвращает код некоторой заданной длины. Длина кода задает точность кодирования, причем близкие друг другу точки будут иметь одинаковый хэш-код. По коду же функция декодирования возвращает координаты и погрешность. Таким образом, хэш-коду соответствует некий квадрат точек в пространстве. Хэш функция кодирования выполняется гораздо быстрее подсчета дистанции, и

для того чтобы исключить соседние точки нам достаточно сравнивать лишь их коды. Коды совпадают - значит они лежат в определенном радиусе. По хэш-коду мы можем получить центр квадрата, и его будем считать локально корректной точкой. Кроме того, при помощи вычисления гео-квадратов можно так же отсеять многие шумы. Для этого всего лишь нужно выбросить из вычисления те квадраты, куда было меньше трех попаданий. Вероятность того, что некорректное значение GPS будет в одном направлении более трех раз невелика, а значит большая часть шумов отсеется.

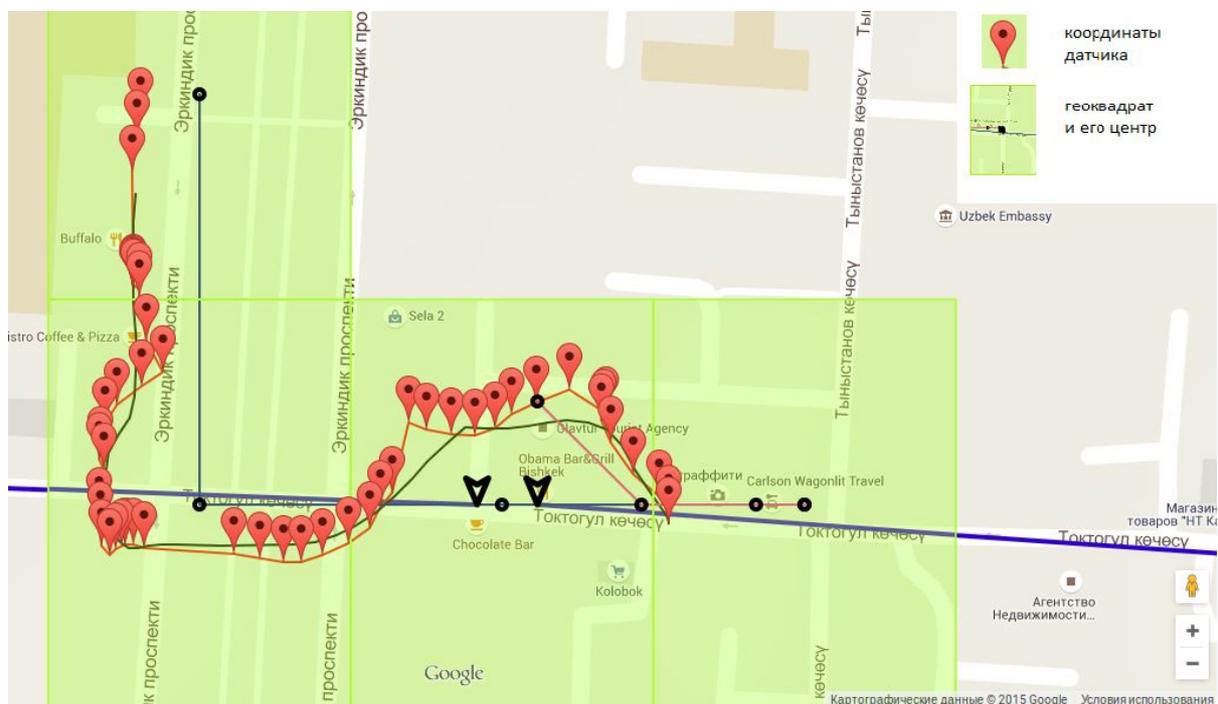


рисунок 5. Иллюстрация работы фильтра гео-квадратов для отсеивания шумов.

Здесь используется допущение, что дороги являются прямыми линиями или локально прямыми, а на перекрестках и поворотах машина на некоторое время сбрасывает скорость. Тогда в идеальном случае в качестве траектории мы получим перекрестки, соединенные прямыми линиями. Возможно, центры квадратов не будут точно совпадать с перекрестками, однако при параллельном переносе длина отрезка не изменяется. Мы взяли точность (количество символов в хэш-коде) 7 - это соответствует квадратам примерно 150 на 112 метров. Такой размер меньше стандартного

городского квартала, а значит дорога определяется почти однозначно. В целом этот метод позволяет значительно выровнять траекторию движения и убрать большинство шумов и дает устойчивую погрешность на одних и тех же маршрутах.

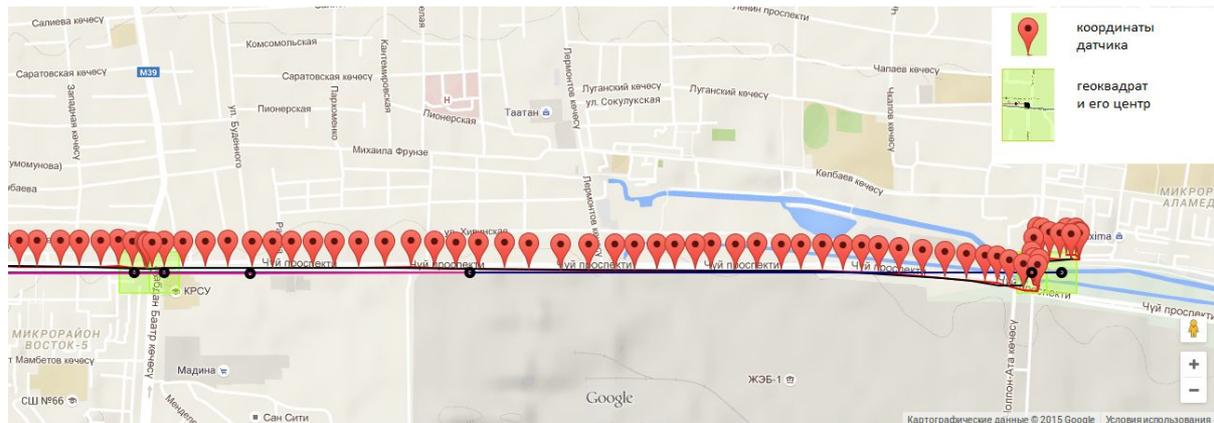


рисунок 6. Иллюстрация фильтра гео-квадратов. Передаются данные только с перекрестков, где скорость была снижена.

Отсев скачков

После фильтра Калмана и фильтра гео-квадратов успешно убирают большинство шумов, и при тестировании подсчет дистанции стал гораздо точнее. По началу этот результат показался достаточным, однако при дальнейшем тестировании и анализе траекторий выяснилось, что на некоторых приборах расстояние все еще слишком велико. Иными словами - либо все считалось более или менее точно, либо критически неправильно - например 400 километров вместо 20. Оказалось, что координата во время поездки смещается в некую точку города, значительно удаленную от реального пути. Иногда сигнал восстанавливается, и возвращает какую-то точку на реальной траектории, и расстояние от ложной точки до дороги суммируется множество раз. Именно отсюда возникают большие накопительного характера ошибки. Что именно в этих особых точках неизвестно - предположительно, GSM вышки или другое устройство, которое может повлиять на сигнал. Идея заранее набрать базу таких плохих точек и считать по умолчанию некорректными оказалась нежизнеспособной, из-за того что каждый раз обнаруживались новые и новые ошибочные точки, куда прыгали

координаты. Таким образом предсказать скачок не удалось, поэтому осталось только постараться распознать. Первым шагом к устранению прыжков стал логический фильтр по скорости. Мы можем посчитать с какой скоростью движется автомобиль от одной точки к другой, и считать точку не корректной, если эта скорость превышает допустимое значение. В начале мы установили этот предел - 140 километров в час, это самая большая скорость движения по магистрали. Но это оказалось значительно выше средней скорости по городу, что привело к интересной ситуации. Иногда точка, куда происходил скачок была в зоне досягаемости с большой скоростью: так как привязки к дорогам нет, виртуальный путь мог проходить сквозь множество зданий. Часть таких случаев удалось отсеять после того, как скоростной предел был изменен на 60 километров в час, так как средняя скорость по городу в большинстве случаев значительно меньше, однако часть все еще осталась из-за того что траектория могла реально быть, но могла оказаться и ошибкой.

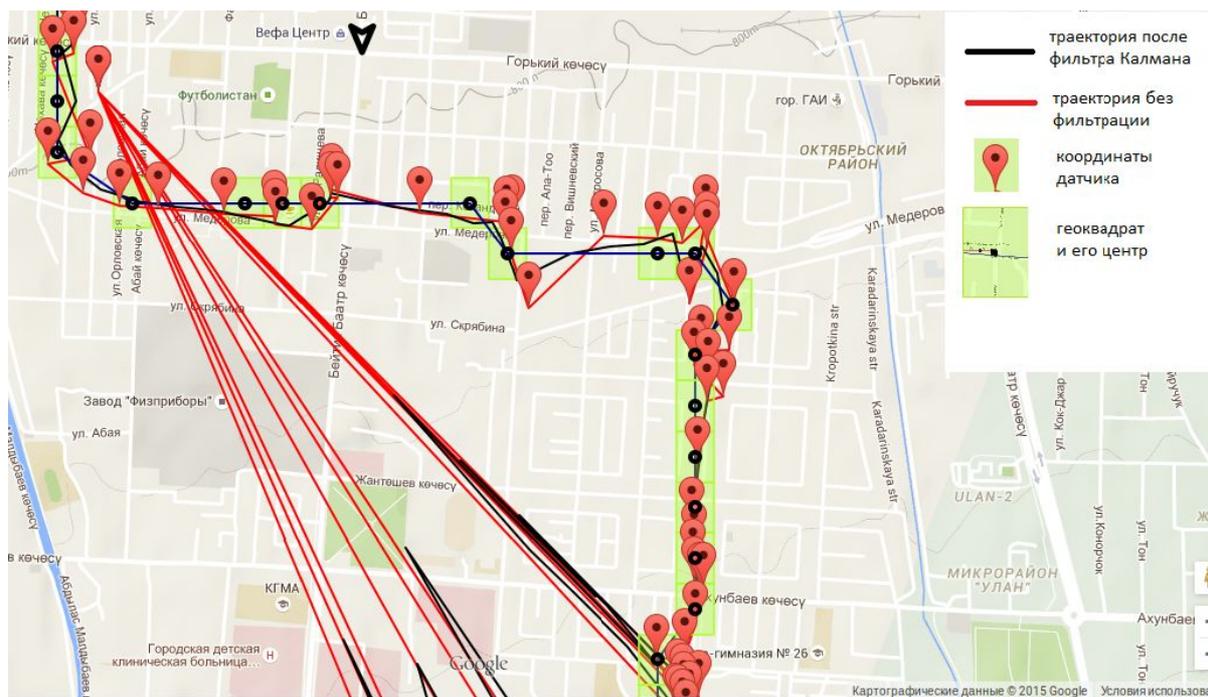


рисунок 7. Иллюстрация устойчивого характера прыжков, а так же накопительного характера ошибки.

Другим методом отсева скачков стала проверка, основанная на характере GPS координат. Как уже упоминалось, даже находясь физически в одной точке, передатчик не будет посылать одни и те же значения много раз подряд. На траекториях это выглядит как “звезды”. Но ситуация с неверными координатами при потере сигнала другая - там ложная точка точна и не изменяется (см. рис. 7). Таким образом, анализируя, сколько раз показатели указывают в одну и ту же точку, мы можем сделать вывод о том является ли она корректной либо ложной. Здесь количество попаданий в точку лучше взять небольшое - даже пяти будет достаточно, поскольку лучше выбросить хорошую точку, чем случайно прибавить большое расстояние. Таким образом, проверка не будет ощутимо задерживать работу алгоритма - максимум проверка займет 5 секунд и не сильно скажется на опыте пользователя.

Тестирование и результаты

Итоговый алгоритм сначала проверяет точки на возможность скачков, затем применяет фильтр Калмана и итоговое расстояние между двумя точками возвращается как расстояние между центрами гео-квадратов. Это позволило добиться приемлимых результатов на всех доступных отчетах о поездках, и больше критически неправильных результатов на “проблемных” телефонах не возникало. В целом удалось добиться устойчивой погрешности до 400 метров.

Приведенный пример (см. рис. 8) наглядно иллюстрируют насколько алгоритм эффективен в фильтрации неправильных координат.

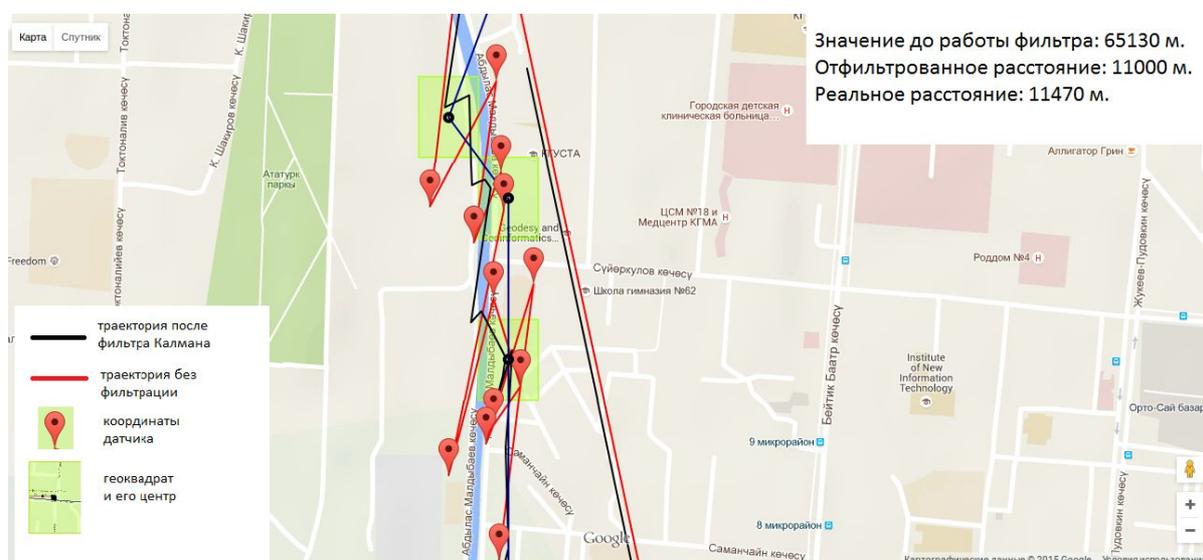


рисунок 8. Пример работы алгоритма фильтрации данных.

Примеры траекторий и результаты работы алгоритма в приведенном таблице ниже.

Неотфильтрованное значение, м.	После фильтра Калмана, м.	После фильтра геоквадратов, м.	Реальное расстояние, м.
65130	29816	11000	11470
226913	79343	8639	5090
19113	16290	16392	16300
24287	25745	23320	23105
8403	7843	8035	7980
9379	8958	9034	9470

таблица 1. Примеры работы алгоритма на различных типах устройств.

По итогам тестирования можно условно разделить группы устройств-приемников gps на три группы - “хорошие”, “средние”, “плохие”. “Хорошие” приемники характеризуются тем, что свое местоположение указывают с достаточной точностью, и алгоритм дает на таких траекториях ухудшение погрешности, так как ошибка привязана к размеру гео-квадрата. В таком случае приходится жертвовать точностью в пользу снижения передаваемой через сеть избыточной информации. Кроме того, это позволяет усреднить картину и выдавать примерно одно и то же расстояние для всех типов устройств.

“Средние” устройства предоставляют свою координату так же с небольшой погрешностью, но с большим количеством шумов. В эту группу отнесены те устройства, для которых алгоритм работает максимально эффективно - то есть отличие результатов работы алгоритма от реального значения дистанции меньше чем разница между сырыми данными и реальным путем. По результатам тестирования, в эту группу попадает большинство траекторий.

“Плохие” устройства стали главной целью в фильтрации координат, так как неотфильтрованные значения давали огромные ошибки. Здесь необходимо было убрать скачки и минимизировать погрешность, так как даже после сглаживания фильтром Калмана траектория была слишком неточной. Здесь становится особенно актуальным метод квадратов, и на траекториях где даже сложно

установить реальный маршрут визуально, результат получается приемлемым. С примерами траекторий разных типов можно ознакомиться в таблице 1.

После работы алгоритма погрешность все еще большая, зачастую она хуже чем у “средних” устройств, но уже не является критически большой.

В среднем же результаты работы алгоритма приближены к реальному расстоянию.

	Неотфильтрованное расстояние, м.	После фильтра Калмана, м.	После фильтра геоквадратов, м.	Реальное расстояние, м.
Среднее	20039,24	11610,6	7546,96	7636,9788
σ	44006,16264	15468,0454	5308,057554	5568,75087
хорошие устройства	9703,833333	8960,333333	8908,666667	9749,166667
σ	6063,207386	5358,011406	5587,641125	6134,598884
средние устройства	8344,461538	7748,461538	7443,923077	7554,230769
σ	6286,897347	6397,221664	5940,658853	5826,26145
плохие устройства	55713,33333	22628,83333	6408,5	5704,078333
σ	79187,96886	26810,38809	3122,02481	3133,474593

таблица 2. Средние значения результатов работы алгоритма для различных типов устройств.

Разделение условно, при разных обстоятельствах одно и то же устройство может переходить из одной категории в другую.

Численные показатели тестирования можно увидеть в приведенной таблице (см. таблицу 2)

Заключение

Был реализован алгоритм, производящий в несколько этапов фильтрацию и уточнение полученных GPS координат.

Минимизированы ошибки всех обнаруженных типов: сглажены шумы, скачки идентифицированы и отсеяны. Таким образом, алгоритм устойчив к кратковременной потере сигнала, и работает с устойчивой погрешностью, при этом выдавая результат почти в реальном времени. Так же метод стабильно работает без задержек на мобильных процессорах, не мешая работе других приложений. Преимущество алгоритма в том, что его можно сразу применять в любом городе мира - так как для корректной работы, в отличие от большинства аналогов не нужен предварительный сбор информации, такой как картография дорог или информация о GSM вышках и wi-fi сетях.

Список литературы

[1] Martin J. D., Krösche J., Boll S. Dynamic GPS-position correction for mobile pedestrian navigation and orientation //WPNC, Workshop on Positioning, Navigation and Communication. – 2006. – С. 199-208.

[2] Cao L., Krumm J. From GPS traces to a routable road map //Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems. – ACM, 2009. – С. 3-12.

[3] Maybeck P. S. The Kalman filter: An introduction to concepts //Autonomous Robot Vehicles. – Springer New York, 1990. – С. 194-204.

[4] Google Maps Roads API

<https://developers.google.com/maps/documentation/roads/intro>

[5] Яндекс Локатор

<https://tech.yandex.ru/locator/>