



А.А. Коротышева, С.Н. Жуков

АЛГОРИТМЫ РЕАЛИЗАЦИИ ТЕХНОЛОГИИ ДОПОЛНЕННОЙ РЕАЛЬНОСТИ В НАВИГАЦИОННОМ ОСНАЩЕНИИ АВТОМОБИЛЬНОГО ТРАНСПОРТА

(Нижегородский государственный университет им. Н.И. Лобачевского)

Введение

В современном мире для построения маршрутов движения водители автомобильного транспорта вместо обычных «бумажных» карт используют навигаторы, предоставляющие географическую информацию о местоположении и основанные на геоинформационных системах (ГИС) [1].

Одним из перспективных направлений в этой сфере является применение технологии дополненной реальности (augmented reality, AR) в виде проекции информации на лобовое стекло автомобиля. Такой способ вывода информации называется технологией HUD (head-up display, проекционный дисплей) [2].

Разработка и применение подобных алгоритмов дополненной реальности совместно с реализацией технологии HUD является актуальной задачей для повышения безопасности дорожного движения. В данной работе предложен и программно реализован алгоритм визуализации объектов с динамическими параметрами, функционально зависящими от геоинформационных данных с использованием сервисов OSRM (Open Street Routing Machine) и Open Street Map [3, 4].

Алгоритм построения маршрута

Основная задача ГИС - построение оптимального маршрута по критерию минимального расстояния или времени. Дорожная сеть, необходимая для поиска, состоит из местных дорог, городских улиц и магистралей и представляется в виде графа, где ребра – это дороги, а вершины – перекрестки, промежуточные и конечные точки. Используемые в ГИС алгоритмы выполняют предварительную обработку графа для ускорения работы отдельных запросов и более эффективного использования памяти. Предварительная обработка графа представляет процесс «сжатия» узлов графа по одному за проход. Чтобы выполнить сжатие, вычисляется кратчайший путь между ближайшими соседями узла, на них размещаются ярлыки и узел помечается как обработанный.

В данной работе для построения маршрутов был выбран маршрутный сервис OSRM – открытый проект с http - сервисом, использующий для оптимизации маршрута эффективный алгоритм Contraction Hierarchies [5]. Обычно в литературе в качестве алгоритмов маршрутизации применяют алгоритмы Дейкстры или A^* [6, 7].

Алгоритм Дейкстры определяется выражением

$$d[v] = \min_{p: u[p]=false} d[p], \quad (1)$$



где $d[v]$ - текущая длина кратчайшего пути из s в v для вершины v , $u[p]$ - булевский массив, в котором сначала все узлы не помечены (значение элементов – false), а на очередной итерации выбирается вершина v с самой маленькой величиной $d[v]$ среди тех, которые ещё не помечены. Выбранная вершина отмечается помеченной, после чего на текущей итерации из вершины v выполняются релаксации — просматриваются все ребра (v, to) , которые исходят из вершины, причем для каждого узла алгоритм пробует улучшить значение $d[to]$:

$$d[to] = \min(d[to], d[v] + len) \quad (2)$$

где len - длина текущего ребра. В конечном итоге после n итераций, все узлы станут помеченными, а алгоритм завершит работу.

Алгоритм A^* определяется выражением

$$f(n) = g(n) + h(n), \quad (3)$$

где $f(n)$ - минимальная стоимость перехода в соседний узел, $g(n)$ - стоимость пути от начальной вершины до любой другой, $h(n)$ - эвристическое приближение стоимости пути от узла n до конечного узла.

Выбранный алгоритм Contraction Hierarchies работает быстрее алгоритмов Дейкстры или A^* , не требует дополнительной памяти, имеет относительно быстрый препроцессинг и состоит из двух стадий. Стадия препроцесса: вершины сортируются в каком-либо жёстком порядке, далее каждая пара рёбер проверяется на предмет её упрощения (возможность заменить эту пару одним ребром). Стадия запроса: с начальной и конечной точки маршрута запускается двусторонний алгоритм Дейкстры с условием, что волны идут только вверх по иерархии (когда они встретятся — путь найден), далее последовательно восстанавливаются сокращённые рёбра.

Расчет расстояния между точками маршрута будем производить по формуле гаверсинусов, чтобы избежать проблем с небольшими расстояниями

$$\Delta\sigma = 2 \arcsin \left\{ \sqrt{\sin^2 \left(\frac{\phi_2 - \phi_1}{2} \right) + \cos \phi_1 \cos \phi_2 \sin^2 \left(\frac{\Delta\lambda}{2} \right)} \right\}, \quad (4)$$

где - $\phi_1, \lambda_1; \phi_2, \lambda_2$ - широта и долгота двух точек в радианах, $\Delta\lambda$ - разница координат по долготу, $\Delta\sigma = \arccos\{\sin \phi_1 \sin \phi_2 + \cos \phi_1 \cos \phi_2 \cos \Delta\lambda\}$ - угловая разница.

Начальный азимут от начальной точки к конечной точке вычислим по формуле

$$\theta = \operatorname{atan2}(\sin \Delta\lambda \cos \phi_2, \cos \phi_1 \sin \phi_2 - \sin \phi_1 \cos \phi_2 \cos \Delta\lambda). \quad (5)$$

Алгоритм визуализации и результаты моделирования

В качестве источника геоинформационных данных при реализации алгоритма визуализации объектов дополненной реальности (рисунок 1) был выбран OpenStreetMap - проект с открытым исходным кодом, которой является аналогом плиточных картографических сервисов, используемых такими системами, как OpenLayers. OpenStreetMap имеет глобальные векторные данные на уровне улиц и других пространственных объектов [4].



При построении маршрута определяются начальная и конечная координаты маршрута, их значения обрабатываются и передаются в маршрутный сервис OSRM. Для определения координат используется приемник GPS спутниковой системы навигации. Скаченные с серверов Open Street Map тайлы объединяются в карту, полученные координаты переводятся в систему координат карты и визуализируются посредством 2D-графики.

Видеопоток с камеры поступает в функцию отрисовки объекта, в которой производится обработка и анализ каждого кадра видеопотока. Вычисляется маска объекта, его растяжение и поворот, что позволяет однозначно задать положение объекта в пространстве. Затем объект выводится на экран при помощи графической библиотеки.

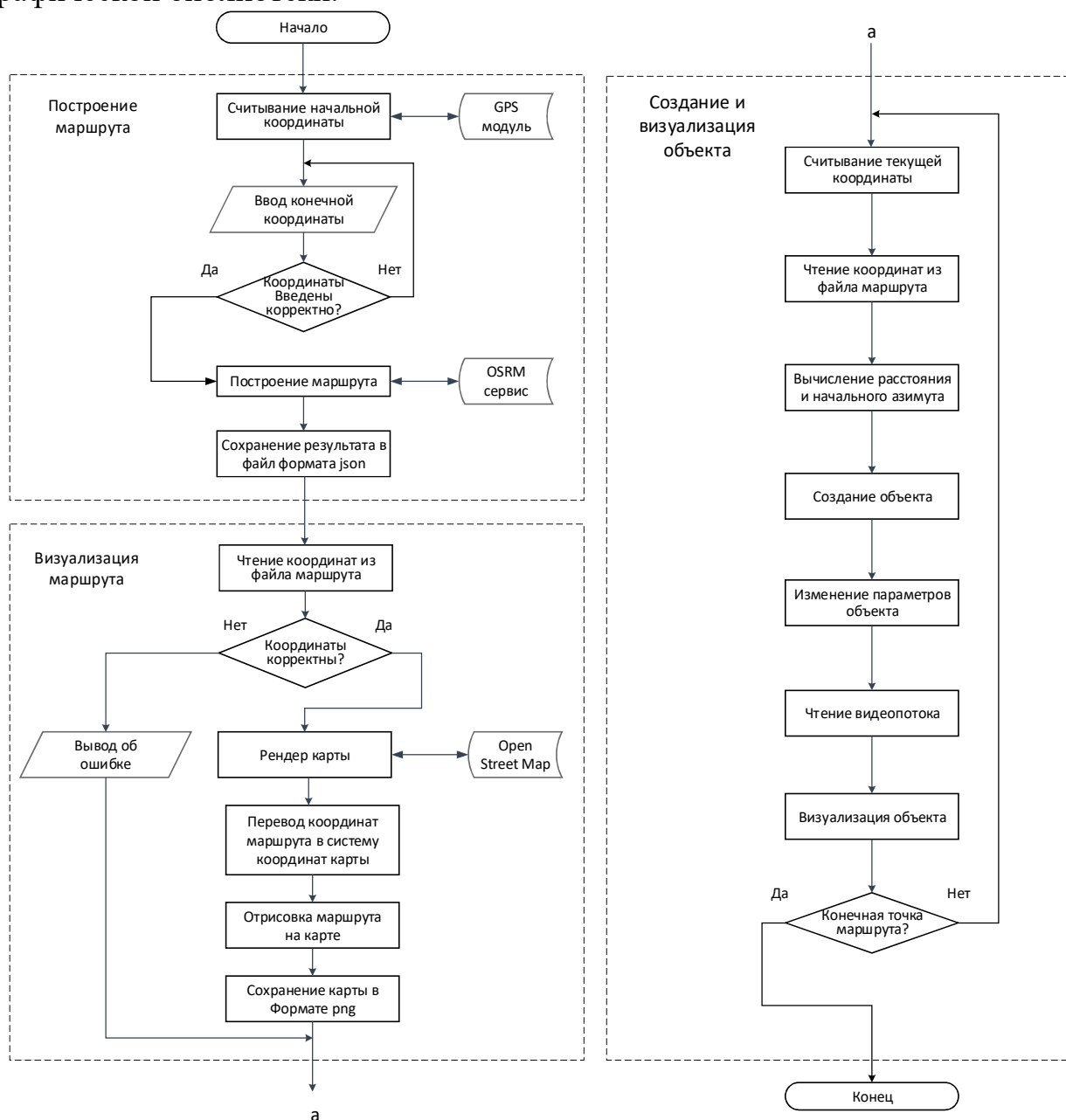


Рис. 1. Алгоритм визуализации объектов дополненной реальности



Разработанный алгоритм был реализован и опробован в программе на языке Python. При проведении моделирования с использованием сервиса OSRM сначала был построен оптимальный граф маршрута (рисунок 2). Затем в режиме реального времени программный код выводил на экран изображение дороги, полученное от видеокamеры, а также дополнительный слой с визуализацией направления движения по заданному маршруту в виде «подсказок» - стрелок-указателей. Маршрут сохранялся в отдельном файле формата json.

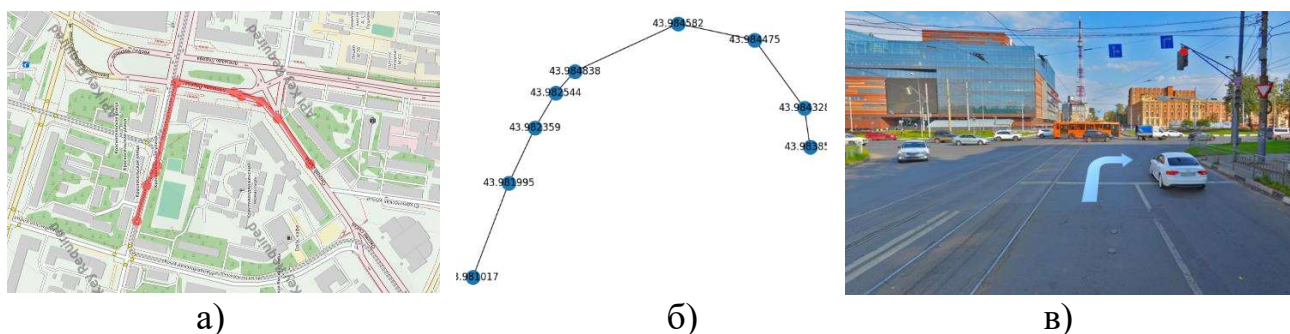


Рис. 2. Результаты моделирования: а) маршрут на карте, б) граф маршрута, в) визуализация объекта дополненной реальности.

Заключение

Создан и опробован алгоритм визуализации объектов с динамическими параметрами, функционально зависящими от геоинформационных данных с использованием сервисов OSRM и Open Street Map. Разработанный интерактивный интерфейс программы визуализации дополненной реальности с проекцией на лобовое стекло автомобиля обладает интегрированным эффектом от совмещения преимуществ навигационных систем и сервисов геоинформационных данных. Предложенный алгоритм может быть использован в программном обеспечении системы навигационного оснащения автомобиля.

Литература

- 1 Капралов, Е.Г. Геоинформатика / Капралов Е., Кошкарев А., Тикунов В., Лурье И., Семин В., Серапинас Б., Сидоренко В., Симонов А. – М.: Academia, 2010.
- 2 Charissis V., Papanastasiou S. Human-machine collaboration through vehicle head up display interface / Cogn Tech Work, 2010. Vol. 12, pp. 41–50.
- 3 OSRM API Documentation [Электронный ресурс] // Project OSRM: [сайт]. URL: <http://project-osrm.org/docs/v5.22.0/api/#general-options> (дата обращения: 31.03.2021).
- 4 OpenStreetMap - wiki-карта мира [Электронный ресурс] // OpenStreetMap: [сайт]. URL: <https://www.openstreetmap.org/> (дата обращения: 31.03.2021).
- 5 Geisberger R., Sanders P., Schultes D., and Delling D. Contraction Hierarchies: Faster and Simpler Hierarchical Routing in Road Networks / Proceedings of



the 7th Workshop on Experimental Algorithms (WEA'08). 5038 of Lecture Notes in Computer Science, 2008, pp. 319–333.

6 Dijkstra E. W. A Note on Two Problems in Connexion with Graphs // Numerische Mathematik, 1959. Vol. 1, pp. 269–271.

7 Hart P. E., Nilsson N. J., Raphael B. A Formal Basis for the Heuristic Determination of Minimum Cost Paths // IEEE Transactions on Systems Science and Cybernetics SSC4, 1968. Vol. 2, pp. 100 - 107.

В.С. Кузьмин, Д.В. Еленев

АВТОМАТИЗИРОВАННАЯ СИСТЕМА МОНИТОРИНГА ТРАНСПОРТНОГО КОМПЛЕКСА ПРЕДПРИЯТИЯ

(Самарский университет)

Транспортный комплекс является важным технологическим объектом для предприятия. Он обеспечивает перевозку грузов в определенные технологические зоны. Данные технологические объекты применяются в различных отраслях: космических, атомных, авиационных и горнодобывающих.

Автоматизированная система управления транспортным комплексом состоит из следующих элементов: программируемые логические контроллеры, панельный компьютер в промышленном исполнении. На рисунке 1 представлена аппаратная структура автоматизированной и вычислительной системы управления [2].

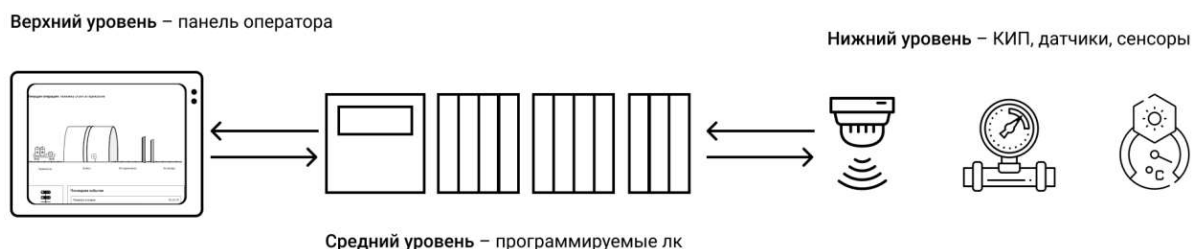


Рисунок 1

С датчиков полевого уровня на программируемый логический контроллер поступают дискретные и аналоговые сигналы с технологического объекта предприятия, после их обработки на программируемом логическом контроллере через интерфейс Ethernet по протоколу modbus TCP обработанные сигналы поступают программные решения панельного компьютера [3].

Основные алгоритмы работы и безопасности важных элементов транспортного комплекса заложены в программируемых логических контроллерах согласно правилам и требованиям основ автоматизированных систем управления технологическими процессами [4]. На рисунке 2 изображены примеры алгоритмов возникновения аварийных ситуаций элементов технологического объекта: выход из строя концевого выключателя, выход из строя реле давления.