

УДК 528.92:004.8

DOI: 10.33764/2411-1759-2021-26-6-96-106

РАЗРАБОТКА МЕТОДИКИ АВТОМАТИЗИРОВАННОЙ АКТУАЛИЗАЦИИ КАРТ ГОРОДСКИХ ТЕРРИТОРИЙ С ПОМОЩЬЮ ТЕХНОЛОГИИ КОМПЬЮТЕРНОГО ЗРЕНИЯ

Никита Вячеславович Порохов

Сибирский государственный университет геосистем и технологий, 630108, Россия, г. Новосибирск, ул. Плеханова, 10, магистрант кафедры картографии и геоинформатики, тел. (999)464-04-58, e-mail: nikitaporoh07@mail.ru

Петр Юрьевич Бугаков

Сибирский государственный университет геосистем и технологий, 630108, Россия, г. Новосибирск, ул. Плеханова, 10, кандидат технических наук, доцент кафедры картографии и геоинформатики, тел. (383)343-18-53, e-mail: peter-bugakov@ya.ru

В настоящее время для обновления карт широко используются спутниковые или аэрофотоснимки. Для ускорения процесса актуализации некоторые крупные картографические сервисы используют системы компьютерного зрения, интеллектуальные системы и другие передовые технологии. Однако частота получения исходных данных – спутниковых снимков – все еще невелика. В связи с этим в статье предлагается методика актуализации карт городских территорий, основанная на получении исходных данных с помощью наземной фотосъемки и дальнейшего распознавания образов зданий и сооружений. В статье обосновывается актуальность темы исследования, приводятся примеры использования технологии распознавания образов для обновления картографических материалов крупнейшими сервисами. Также рассматриваются основные технические характеристики программного и аппаратного обеспечения, необходимого для реализации эксперимента по предлагаемой методике.

Ключевые слова: автоматизация, открытые карты, компьютерное зрение, нейронные сети

Введение

Современные открытые картографические сервисы, такие как Викимапия, Яндекс.Народная карта и OpenStreetMap (OSM), дают всем зарегистрированным пользователям возможность редактирования объектов в реальном времени с последующим отображением изменений на карте. На данный момент самым популярным проектом является OpenStreetMap. О его популярности говорит то, что крупные веб-картографические проекты используют лицензию OSM для своих нужд. В первую очередь, данные с OSM использует сервис 2ГИС, который является одним из крупнейших поставщиков веб-карт в России и ближнем зарубежье наравне с Яндекс.Картами.

OpenStreetMap предоставляет простой интерфейс для редактирования картографических данных, а также позволяет скачивать необходимый фрагмент (тайл) карты в формате .osm или последнюю полную копию базы данных OpenStreetMap в формате XML.

Применение новых разработок в области интеллектуальных систем для актуализации карт позволяет:

- увеличить скорость сбора и обработки исходных данных;
- автоматизировать процесс обновления информации;
- уменьшить трудозатраты на обновление карт.

Современные технологии дистанционного зондирования уже повсеместно используются при решении похожих задач. Как правило, для обновления карт используются данные, полученные в результате спутниковой или аэрофотосъемки. Такой способ актуализации карт имеет существенный недостаток – он может занимать достаточно большой промежуток времени.

Одним из основных источников данных для редактирования карты в OSM служат спутниковые снимки Esri. Если судить по слою World Imagery с сайта Esri, то последнее обновление снимков для территории Новоси-

бирска проводилось 30.06.2020. Также существует проблема корректной идентификации объектов по данным спутниковой съемки. Исследуемый объект могут закрыть кроны деревьев или иные помехи. Таким образом, некоторые данные, отображенные на карте OSM, требуют актуализации с использованием других более оперативных подходов с использованием технологии компьютерного зрения.

Методы и материалы

В настоящее время крупнейшим картографическим сервисам Google Map и Яндекс.Карты приходится ежедневно обрабатывать огромное количество данных для поддержания своих карт в актуальном состоянии. Существенно повысить скорость обработки данных позволяют современные

технологии, такие как компьютерное зрение, большие данные, искусственный интеллект и т. д. [1–11].

Например, компания Яндекс, чтобы облегчить работу картографам, начала применять технологию компьютерного зрения для отрисовки зданий на карте. Их подход заключается в следующем. На вход системы поступает спутниковый снимок, на котором с помощью процесса семантической сегментации определяется принадлежность каждой точки зданию [2]. На территориях, где здания стоят на достаточном расстоянии друг от друга, например, сельская местность, сразу осуществляется переход к следующему этапу – векторизации. Однако на территориях с плотной застройкой дома склеиваются в связную область, которую необходимо разделить (рис. 1).

Для разделения областей на отдельные объекты производится поиск ребер (рис. 2).



Рис. 1. Результат семантической сегментации



Рис. 2. Поиск границ домов

Следующим этапом является векторизация найденных границ.

У данного способа есть несколько достоинств:

- выполнение большого объема работ за короткий промежуток времени. Если появляется новый спутниковый снимок с большим количеством неадресованных объектов, то с помощью данной технологии можно достаточно быстро нанести новые объекты на карту;

- способность работать с разным типом застройки.

К недостаткам относят:

- большой временной промежуток между обновлениями спутниковых снимков, по которым происходит отрисовка зданий;

- крыша здания может быть скрыта высокой кроной дерева, что будет препятствовать распознаванию, и в дальнейшем будет необходимо проводить рекогносцировку объекта на местности.

На основе анализа выявленных достоинств и недостатков существующего способа актуализации карт предлагается методика, которая оперирует данными, полученными с помощью наземной съемки и специального оборудования.

Основная идея методики заключается в выявлении новых или снесенных зданий на основе анализа фотоизображений, сделанных камерой, которая установлена на крыше автомобиля. Также методика будет позволять автоматически отмечать на карте положение изменений, выявленных на местности.

Рассмотрим некоторые технические особенности аппаратных и программных средств, необходимых для реализации разрабатываемой методики. Для непосредственной съемки необходима камера. Выбор угла обзора объектива камеры должен основываться на расстоянии до здания и размерах его самой узкой части. При проведении съемки может возникнуть ситуация, когда в поле зрения камеры попадает торцевая сторона дома. Стандартная ширина типового жилого дома варьируется в пределах 13–14 м. Если дорога, по которой движется автомобиль с оборудованием, имеет одну полосу для движения, и расстояние от проезжей части составляет по-

рядка 10–15 м, то камера должна запечатлеть на снимке дом, который расположен к камере торцом. Например, на ул. Плахотного расстояние от проезжей части до жилых домов составляет в среднем 12 м. Необходимо подобрать угол обзора камеры таким образом, чтобы на указанном расстоянии торец дома попал на снимок. Путем простых вычислений можно определить, что наиболее подходящим будет угол обзора объектива камеры, равный 60° . В этом случае при расстоянии 12 м до объекта съемки поле зрения фотокамеры будет составлять 13,85 м. Этого вполне хватает, чтобы торец дома полностью был в кадре.

Также необходимы лазерный дальномер для измерения расстояния от фотокамеры до здания, GPS-приемник для получения координат регистрирующих приборов, а также электронный компас для определения значения дирекционного угла с целью определения координат здания. Единственным требованием к данному оборудованию является возможность удаленного управления и передачи данных посредством подключения к ноутбук [12–16].

Для реализации технологии компьютерного зрения, как правило, используют программные библиотеки OpenCV и TensorFlow. OpenCV – это библиотека компьютерного зрения и машинного обучения с открытым исходным кодом. В нее входят более 2 500 алгоритмов, в которых есть как классические, так и современные алгоритмы для компьютерного зрения и машинного обучения.

TensorFlow – это комплексная платформа с открытым исходным кодом для машинного обучения. Она имеет комплексную гибкую экосистему инструментов, библиотек и ресурсов сообщества, которая позволяет исследователям продвигать новейшие достижения в области машинного обучения, а разработчикам – легко создавать и развертывать приложения на основе машинного обучения [17–21].

Для хранения фотоснимков, пространственных и семантических данных предлагается использовать СУБД PostgreSQL, поскольку она поддерживает работу с модулем PostGIS [22–25].

В решении задачи распознавания образов лучше всего себя зарекомендовала сверточная нейронная сеть, реализованная с использованием программных модулей обработки данных на языке Python [1–15, 26]. Благодаря операции свертки, при которой каждый фрагмент изображения умножается на матрицу свертки поэлементно, сеть становится глубже и лучше выполняет анализ входных данных.

Чтобы нейронная сеть была способна идентифицировать здание среди множества объектов улицы, необходимо ее обучить с помощью набора данных (Dataset). В сети Интернет существует множество наборов данных, как платных, так и бесплатных. Главная проблема заключается в том, что все они содержат изображения объектов зарубежной архитектуры, которая существенно отличается от российской [9–11, 27, 28].

В связи с этим необходимо сначала создать набор данных для обучения нейронной сети, который будет содержать характерную для нашего региона архитектуру. Создание набора данных включает в себя сбор и структурирование фотоизображений, а также разметку изображенных на них объектов.

После подготовки тренировочного набора данных происходит обучение нейронной сети. Чтобы узнать эффективность нейронной сети, нужно протестировать ее на большом количестве изображений.

Результаты разработки методики

Процесс актуализации открытых карт заключается в распознавании зданий на фотоснимках, полученных при наземной съемке улицы, с последующим вычислением координат здания и отображением его на карте OSM (если он не был нанесен на карту ранее). На рис. 3 представлена схема, показывающая этапы процесса актуализации.

На автомобиль устанавливается оборудование, состоящее из камеры, GPS-приемника, дальномера и цифрового компаса. Автомобиль едет по заранее заданному маршруту, а камера ведет непрерывную съемку прилегающих к улице объектов городской инфраструктуры. Данные с камеры поступают на

вход нейронной сети, которая классифицирует все объекты на статическом снимке или видеокadre. Это делается для того, чтобы идентифицировать здание из множества объектов, которые не нужны для дальнейшей обработки.

Из множества объектов, изображенных на снимке, необходимо выбрать здание; для этого все объекты классифицируются одним из следующих способов:

1) перечисление. Каждый класс задается путем прямого указания его членов;

2) задание общих свойств. Класс задается указанием некоторых признаков, присущих всем его членам;

3) кластеризация. В случае, когда объекты описываются векторами признаков или измерений, класс можно рассматривать как кластер.

Первый этап актуализации заключается в съемке придорожной территории и сборе данных, полученных от камеры, дальномера, GPS-приемника и компаса (см. рис. 3, блоки под номером 1). Все оборудование крепится на штативе и располагается на крыше автомобиля. Перед началом съемки составляется маршрут следования транспорта. Основную проблему при съемке городских улиц может составить пассажирский транспорт.

Как правило, он движется по крайней правой (выделенной) полосе и может перекрывать луч дальномера. Камера и дальномер должны располагаться на относительно большой высоте, чтобы объекты, находящиеся между дальномером и зданием, не перекрывали луч лазера.

Для расчета высоты штатива возьмем один из самых распространенных в Новосибирске автобусов НефАЗ. Его высота составляет 3,4 м. Высота легкового автомобиля составляет 1,6 м. Из этого следует, что штатив для оборудования должен быть высотой около 2 м, чтобы луч дальномера проходил над большей частью транспортных средств.

Визирный луч камеры и луч лазера дальномера должны быть сориентированы параллельно и направлены строго перпендикулярно вектору движения автомобиля.

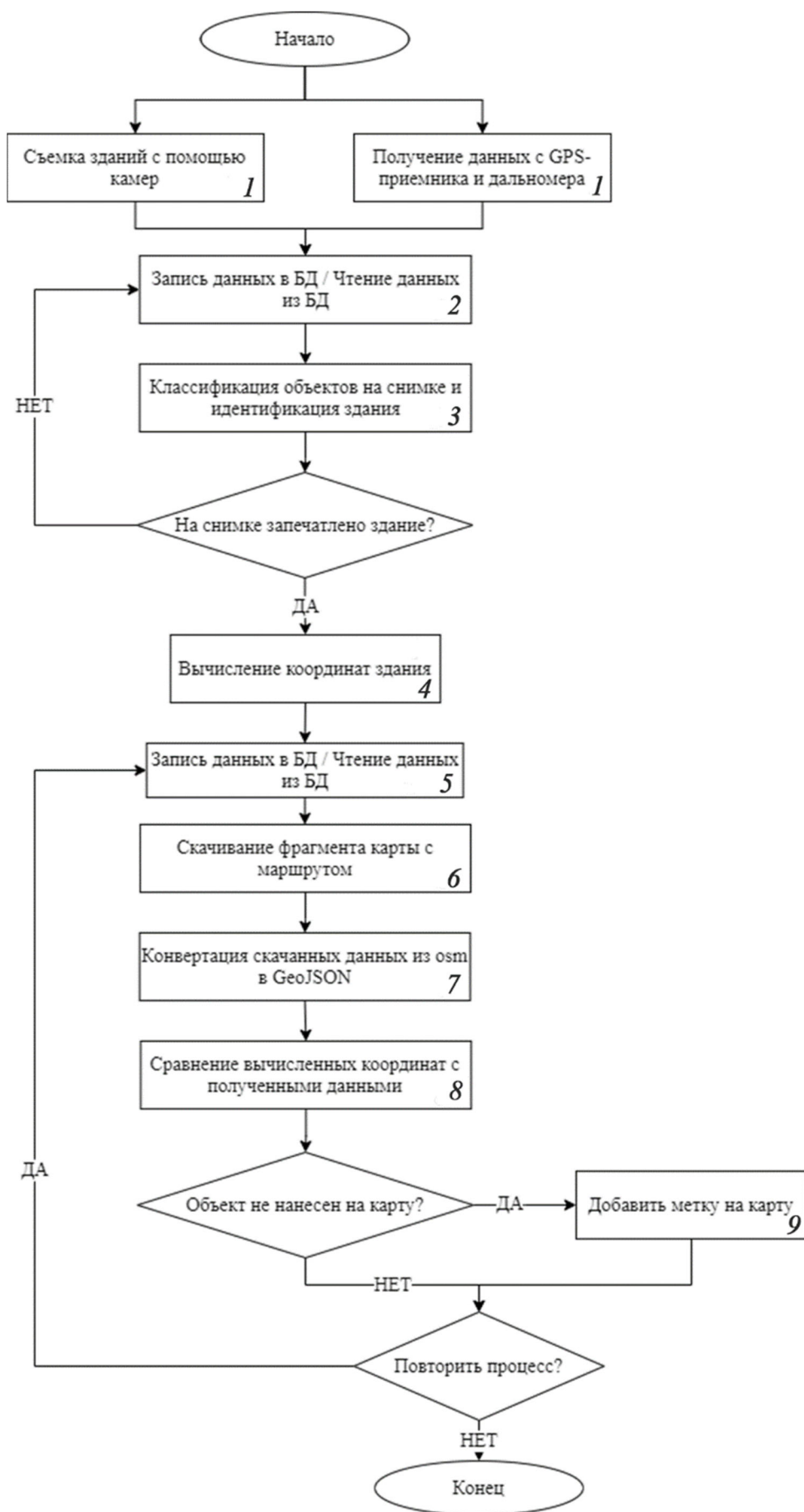


Рис. 3. Схема разрабатываемого метода актуализации

Схема направления движения автомобиля и направления съемки представлена на рис. 4.

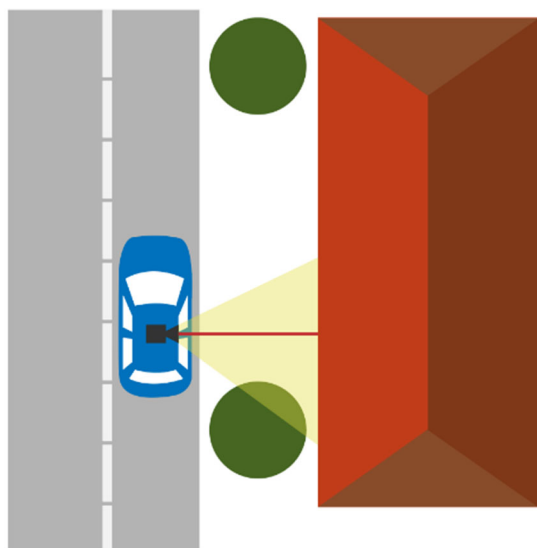


Рис. 4. Схема съемки

Так как в городе автомобиль движется с переменной скоростью за-за светофоров, участков с ограничением скорости и других препятствий, съемка должна происходить дискретно. GPS-трекер отслеживает движение автомобиля и через каждые 5 м сохраняет значение координат, а также передает сигнал на камеру для выполнения съемки. В свою очередь дальномер передает на компьютер (ноутбук) расстояние, а компас – азимут.

Размер изображения, получаемый с камеры, не обязательно должен быть большим – достаточно, чтобы человек смог обнаружить на нем необходимый объект. Также небольшой размер обусловлен ограничением памяти для хранения полученных изображений. Оптимальным размером снимка был выбран формат 255×255 пикселей. Статическое изображение такого размера занимает порядка 30 КБ памяти на цифровом носителе.

В процессе прохождения маршрута все собранные данные сохраняются в базу данных (см. рис. 3, блок 2). Впоследствии лишние снимки, не содержащие изображение зданий, необходимо удалить. Эту задачу предлагается автоматизировать с использованием технологии распознавания объектов на изображении. Таким образом, после этапа сбора данных на вход нейронной сети начинают посту-

пать полученные снимки. Она будет классифицировать каждый объект, который был запечатлен на изображении (рис. 3, блок 3). Анализируя придорожную территорию, можно выделить классы объектов, которые чаще всего могут попасть на снимок: автомобили, растительность, торговые павильоны и т. д. Как только все объекты будут классифицированы, среди них необходимо идентифицировать здания. Если здание на снимке отсутствует, то он удаляется из базы данных.

Когда здание на снимке обнаружено, можно приступить к вычислению координат точки, принадлежащей ему (см. рис. 3, блок 4). При этом точка может быть выбрана произвольная, например, центральная на снимке, при условии, что изображение здания занимает более половины площади снимка. Нахождение координат искомой точки на основе координат съемочного оборудования, длине линии и углу между ними осуществляется путем решения прямой геодезической задачи. При этом вычисленные координаты точки необходимо перевести в географическую систему координат.

Как только нейронная сеть обработает все изображения, а программный модуль вычислит координаты точек зданий, данные запишутся в базу данных (см. рис. 3, блок 5). Этот процесс служит для того, чтобы следующий блок сравнения координат обрабатывал весь объем уже готовых данных, а не ожидал каждую запись по отдельности.

Примерная структура таблицы базы данных:

- отметка времени, когда было получено изображение;
- GPS-координаты автомобиля;
- расстояние от автомобиля до здания;
- вычисленные координаты здания;
- изображение с уже классифицированным зданием.

После того, как этап съемки будет завершен, должен быть подготовлен фрагмент карты (тайл), отображающий маршрут движения автомобиля с оборудованием (см. рис. 3, блок 6). Далее выполняется скачивание информации по всем объектам, отображенным на тайле, в формате .osm. Так как этот формат не очень удобен для работы, его необходимо

конвертировать в формат .json или .geojson (см. рис. 3, блок 7). Данные, скачанные с OSM, имеют теги, наличие которых облегчает поиск необходимых объектов и их координат, в частности, зданий (building). При конвертации в формат .geojson всем объектам присваиваются реальные координаты вместо ссылок на эти координаты в базе OSM [29].

На следующем этапе рассчитанные координаты предполагаемого здания передаются на вход программного модуля, который определяет, нанесен ли объект на карту (см. рис. 3, блок 8). Для реализации модуля необходим доступ к данным OpenStreetMap с целью сравнения координат. Чтобы определить, попадает ли точка с вычисленными координатами в область здания, вокруг нее строится буферная зона. В случае если она перекрывает точку с известными координатами, помеченную тегом building или находится внутри полигона, построенного по скачанным точкам, то она пропускается. Это означает, что объект уже нанесен на карту и нет необходимости инфор-

мировать пользователей OpenStreetMap, что объект отсутствует.

В противном случае, если буферная зона вокруг исследуемой точки не перекрывает ни полигон, ни какие-то другие точки, то ставится отметка, информирующая, что объект отсутствует на карте.

На завершающем этапе идет процесс постановки метки на карту (см. рис. 3, блок 9). Данная метка служит для информирования пользователей о том, что объект отсутствует на карте или был снесен (рис. 5). Метка содержит информацию о пользователе, который создал ее, уникальный идентификатор, сколько времени прошло с даты создания, географические координаты и описание. В описание можно будет добавлять дополнительную информацию, которая в дальнейшем поможет внести корректировки в карту. Например, можно добавлять ссылку на изображение, которое было получено во время съемки. Это даст пользователю более полное представление об объекте.

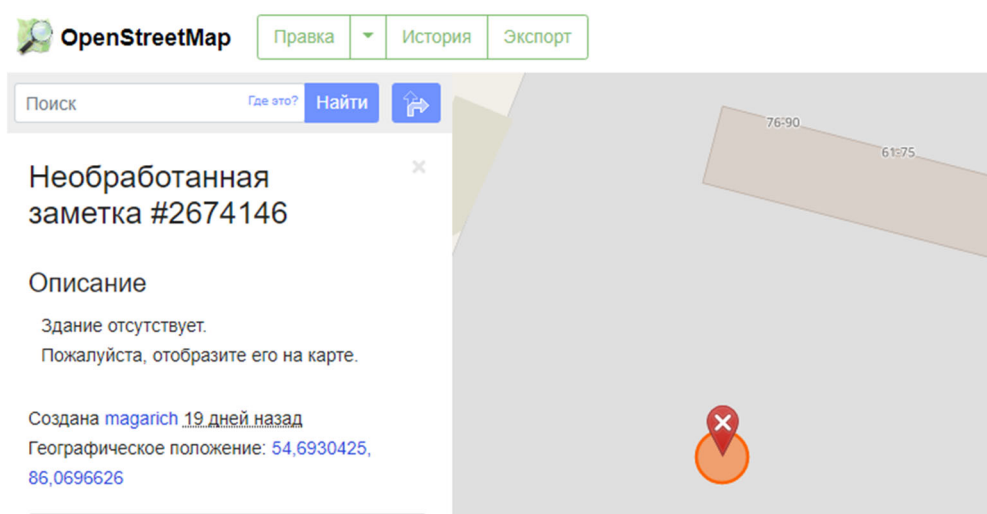


Рис. 5. Пример метки

Заключение

В статье была описана основа методики автоматизированной актуализации открытых карт городских территорий с помощью технологии компьютерного зрения. Также были рассмотрены основные технические характеристики программного и аппаратного обеспечения, необходимого для реализации экспе-

римента по предлагаемой методике. На данном этапе исследования апробация методики выполнялась частично. Были проведены эксперименты по тренировке сверточной нейронной сети на ограниченном наборе размеченных фотографий зданий и сооружений, а также ее использованию для распознавания зданий по фото- и видеоматериалу. Также были написаны фрагменты программного

кода на языке Java для взаимодействия с программным интерфейсом (API) OSM для автоматизации нанесения метки отсутствующего или снесенного здания. Лабораторные эксперименты показали ключевую возможность реализации автоматизированной системы актуализации карт городских территорий с помощью технологии компьютерного зрения на основе разработанной методики. Полноценная апробация методики требует:

– подготовки набора данных для машинного обучения, содержащего размеченные фотографические изображения зданий Но-

восибирска мощностью около 2 000 элементов;

– изготовления штанги и кронштейна для крепления съемочного оборудования на крыше автомобиля;

– написания программного обеспечения для автоматизации процесса сбора большого объема данных.

Дальнейшее развитие и последующее внедрение данной методики, по мнению авторов, может способствовать увеличению скорости и снижению стоимости процесса актуализации карт городских территорий.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Ляшов М. В., Береза А. Н., Бабаев А. М., Коцюбинская С. А. Нейросетевая система отслеживания и распознавания объектов в видеопотоке // Современные наукоемкие технологии. – 2018. – № 12-1. – С. 102–107.
2. Ямашкин А. А., Ямашкин С. А. Использование нейронных сетей прямого распространения для ландшафтного картографирования на базе космических снимков // Геодезия и картография. – 2014. – № 11. – С. 52–58.
3. Кирпичников А. П., Ляшева С. А., Трегубов В. М., Шильников К. В., Шлеймович М. П. Обнаружение трещин дорожного покрытия на изображениях // Вестник технологического университета. – 2019. – Т. 22, № 8. – С. 186–190.
4. Локтев А. А., Алфимцев А. Н., Локтев Д. А. Алгоритм распознавания объектов // Вестник МГСУ. – 2012. – № 5. – С. 194–201.
5. Буэно Г. Г., Эспиноса А. Х. Л., Суарес О. Д. Обработка изображений с помощью OpenCV. – М. : ДМК-Пресс, 2016. – 210 с.
6. Прохоренок Н. А. OpenCV и Java. Обработка изображений и компьютерное зрение. – СПб. : БХВ-Петербург, 2018. – 320 с.
7. Булыгин Д. А., Мамонова Т. Е. Распознавание жестов рук в режиме реального времени // Научный вестник НГТУ. – 2020. – № 1(78). – С. 25–40.
8. Тарасов А. В. Современные методы оперативного картографирования нарушений лесного покрова // Вестник СГУГиТ. – 2020. – Т. 25, № 3. – С. 201–213.
9. Li J., Allinson N. M. Subspace learning-based dimensionality reduction in building recognition // Neurocomputing. – 2009. – Vol. 73 (1–3). – P. 324–330.
10. Li J., Allinson N. M. Building recognition in urban environments: A survey of state-of-the-art and future challenges // Information Sciences. – 2014. – Vol. 277. – P. 406–420.
11. Зеленцов И. А. Распознавание образов. Обзорная лекция [Электронный ресурс]. – Режим доступа: http://it-claim.ru/Persons/Zelencov/Lecture_presentation.pdf (дата обращения 22.10.2020).
12. ГОСТ 21830–76. Приборы геодезические. Термины и определения. – М. : Изд-во стандартов, 1986.
13. ГОСТ Р 52928–2010. Система спутниковая навигационная глобальная. Термины и определения. – М. : Стандартинформ, 2018.
14. ГОСТ Р 53864–2010. Глобальная навигационная спутниковая система. Сети геодезические спутниковые. Термины и определения. – М. : Стандартинформ, 2018.
15. ГОСТ 32453–2017. Глобальные навигационные спутниковые системы. Системы координат. Методы преобразований координат определяемых точек. – М. : Стандартинформ, 2017.
16. ГОСТ Р ИСО 17123–8–2001. Государственная система обеспечения единства измерений. Оптика и оптические приборы. Методы полевых испытаний геодезических и топографических приборов. Часть 8. Полевые испытания GNSS-аппаратуры в режиме «Кинематика в реальном времени». – М. : Стандартинформ, 2019.

17. TensorFlow [Электронный ресурс]. – Режим доступа: <https://www.tensorflow.org/> (дата обращения: 01.12.2020).
18. Ермаков М. К., Вартанов С. П. Подход к проведению динамического анализа java-программ методом модификации виртуальной машины Java // Труды ИСП РАН. – 2015. – Т. 27, № 2. – С. 23–38.
19. Oracle Java [Электронный ресурс]. – Режим доступа: <https://www.oracle.com/ru/java/> (дата обращения: 07.12.2020).
20. Брюс Эккель. Философия Java. – 4-е изд. – СПб. : Питер, 2018. – 1168 с.
21. TechCrunch [Электронный ресурс]. – Режим доступа: <https://techcrunch.com/> (дата обращения: 03.03.2021).
22. Пантелимонов М. В., Бучацкий Р. А., Жуйков Р. А. Кэширование машинного кода в динамическом компиляторе SQL-запросов для СУБД PostgreSQL // Труды ИСП РАН. – 2020. – Т. 32, № 1. – С. 205–220.
23. PostgreSQL [Электронный ресурс]. – Режим доступа: <https://www.postgresql.org/> (дата обращения: 03.12.2020).
24. Sharayeu Y. Machine learning and lambda architecture methods application to optimize PostgreSQL database performance // Big data and advanced analytics. – 2019. – № 5. – С. 81–82.
25. ArcMap [Электронный ресурс]. – Режим доступа: <https://desktop.arcgis.com/ru/arcmap/> (дата обращения 10.02.2021).
26. Коэльё Л. П., Ричерт В. Построение систем машинного обучения на языке Python. – Пер. с англ. – М. : ДМК Пресс, 2015. – 303 с.
27. Philbin J., Chum O., Isard M., Sivic J., Zisserman A. Object retrieval with large vocabularies and fast spatial matching // 2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007) (18–23 June 2007). – Minneapolis, Minnesota, USA. doi: 10.1109/CVPR.2007.383172.
28. Shao T. S. H., Gool L. V. Zubud-Zurich buildings database for image based recognition : Technique Report № 260. – Swiss Federal Institute of Technology, 2003.
29. GEOJSON [Электронный ресурс]. – Режим доступа: <https://geojson.org/> (дата обращения: 22.01.2021).

Получено 26.04.2021

© Н. В. Порохов, П. Ю. Бугаков, 2021

DEVELOPMENT OF A METHODOLOGY FOR AUTOMATED UPDATING MAPS OF URBAN AREAS USING COMPUTER VISION TECHNOLOGY

Nikita V. Porokhov

Siberian State University of Geosystems and Technologies, 10, Plakhotnogo St., Novosibirsk, 630108, Russia, Graduate, Department of Cartography and Geoinformatics, phone: (999)464-04-58, e-mail: nikitaporoh07@mail.ru

Petr Yu. Bugakov

Siberian State University of Geosystems and Technologies, 10, Plakhotnogo St., Novosibirsk, 630108, Russia, Ph. D., Associate Professor, Department of Cartography and Geoinformatics, phone: (383)343-18-53, e-mail: peter-bugakov@yandex.ru

Currently, satellite or aerial photographs are widely used to update maps. To speed up the updating process, some large cartographic services use computer vision systems, intelligent systems and other advanced technologies. However, the frequency of obtaining the initial data-satellite images-is still low. In this regard, the article proposes a method for updating maps of urban areas, based on obtaining initial data using ground photography and further recognition of images of buildings and structures. The article substantiates the relevance of the research topic, provides examples of the use of image recognition technology for updating cartographic materials by the largest services. The main technical characteristics of the software and hardware necessary for the implementation of the experiment according to the proposed method are also considered.

Keywords: automation, open maps, computer vision, neural networks

REFERENCES

1. Lyashov, M. V., Bereza, A. N., Babaev, A. M., & Kotsyubinskaya, S. A. (2018). Object tracking and recognition system based on neural network in a video stream. *Sovremennye naukoemkie tekhnologii [Modern High Technologies]*, 12-1, 102–107 [in Russian].
2. Yamashkin, A. A., & Yamashkin, S. A. (2014). Using neural networks of direct propagation for landscape mapping based on space images. *Geodeziya i kartografiya [Geodesy and Cartography]*, 11, 52–58 [in Russian].
3. Kirpichnikov, A. P., Lyasheva, S. A., Tregubov, V. M., Shilnikov, K. V., & Shlyemovich, M. P. (2019). Detecting pavement cracks in images. *Vestnik tekhnologicheskogo universiteta [Vestnik of Technological University]*, 22(8), 186–190 [in Russian].
4. Loktev, A. A., Alfimtsev, A. N., & Loktev, D. A. (2012). Algorithm of object recognition. *Vestnik MGSU*, 5, 194–201 [in Russian].
5. Bueno, G. G., Espinosa, A. Kh. L., & Soares, O. D. (2016). *Obrabotka izobrazheniy s pomoshch'yu OpenCV [Learning Image Processing with OpenCV]*. Moscow: DMK-Press Publ., 210 p. [in Russian].
6. Prokhorenok, N. A. (2018). *OpenCV and Java. Obrabotka izobrazheniy i komp'yuternoe zrenie [OpenCV and Java. Image processing and computer vision]*. Saint Petersburg: BHV-Peterburg, 320 p. [in Russian].
7. Bulygin, D. A., & Mamonova, T. E. (2020). Recognition of hand gestures in real time. *Nauchnyy vestnik NGTU [Science Bulletin of the NSTU]*, 1(78), 25–40 [in Russian].
8. Tarasov, A. V. (2020). Traditional and modern methods of satellite images processing for operational mapping of forest cover disturbances. *Vestnik SGUGiT [Vestnik SSUGT]*, 25(3), 201–213 [in Russian].
9. Li, J., & Allinson, N. M. (2009). Subspace learning-based dimensionality reduction in building recognition. *Neurocomputing*, 73(1–3), 324–330.
10. Li, J., & Allinson, N. M. (2014). Building recognition in urban environments: A survey of state-of-the-art and future challenges. *Information Sciences*, 277, 406–420.
11. Zelentsov, I. A. (n. d.). Pattern recognition. Overview lecture. Retrieved from http://it-claim.ru/Persons/Zelencov/Lecture_presentation.pdf (accessed October 22, 2020).
12. GOST 21830–76. Geodetic instruments. Terms and definitions. Moscow: Standards Publ., 1986 [in Russian].
13. Standards Russian Federation. (2018). GOST R 52928–2010. Global navigation satellite system. Terms and definitions. Moscow: Standardinform Publ. 2018 [in Russian].
14. Standards Russian Federation. (2018). GOST R 53864–2010. Global navigation satellite system. Geodetic satellite frames. Terms and definitions. Moscow: Standardinform [in Russian].
15. Standards Russian Federation. (2017). GOST 32453–2017. Global navigation satellite system. Coordinate systems. Methods of transformations for determined points coordinate. Moscow: Standardinform [in Russian].
16. Standards Russian Federation. (2019). GOST R ISO 17123-8–2011. State system for ensuring the uniformity of measurements. Optics and optical instruments. Field procedures for testing geodetic and surveying instruments. Part 8. GNSS field measurement systems in real-time kinematic (RTK). Moscow: Standardinform [in Russian].
17. TensorFlow. (n. d.). Retrieved from <https://www.tensorflow.org/> (accessed December 01, 2020).
18. Ermakov, M., & Vartanov, S. (2015). Dynamic Java program analysis using virtual machine modification. *Trudy ISP RAN [Proceedings of ISP RAS]*, 27(2), 23–38 [in Russian].
19. Oracle Java. (n. d.). Retrieved from <https://www.oracle.com/ru/java/> (accessed December 07, 2020).
20. Bruce Eckel. (2018). *Filosofiya Java [Java philosophy]* (4th ed.). Saint Petersburg: Piter Publ., 1168 p. [in Russian].
21. TechCrunch. (n. d.). Retrieved from <https://techcrunch.com/> (accessed March 03, 2021).
22. Pantilimonov, M. V., Buchatskiy, R. A., & Zhuykov, R. A. (2020). Machine code caching in PostgreSQL query JIT-compiler. *Trudy ISP RAN [Proceedings of ISP RAS]*, 32(1), 205–220 [in Russian].
23. PostgreSQL. (n. d.). Retrieved from <https://www.postgresql.org/> (accessed December 03, 2020).
24. Sharayeu, Y. (2019). Machine learning and lambda architecture methods application to optimize PostgreSQL database performance. *Big Data and Advanced Analytics*, 5, 81–82.
25. ArcMap. (n. d.). Retrieved from <https://desktop.arcgis.com/ru/arcmap/> (accessed February 10, 2021).

26. Coeljo, L. P., & Richert, V. (2015). *Postroenie sistem mashinnogo obucheniya na yazyke Python [Construction of machine learning systems in Python]*. Moscow: DMK Press Publ., 303 p. [in Russian].
27. Philbin, J., Chum, O., Isard, M., Sivic, J., & Zisserman, A. (2007). Object retrieval with large vocabularies and fast spatial matching. *2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007)*. Minneapolis, Minnesota, USA. doi: 10.1109/CVPR.2007.383172.
28. Shao, T. S. H., & Gool, L. V. (2003). Zubud-Zurich buildings database for image based recognition. Technique Report No. 260. Swiss Federal Institute of Technology.
29. GEOJSON. (n. d.). Retrieved from <https://geojson.org/> (accessed January 22, 2021).

Received 26.04.2021

© N. V. Porokhov, P. Yu. Bugakov, 2021