

## **АЛГОРИТМ РАЗРАБОТКИ АВТОМАТИЗИРОВАННОЙ АКТУАЛИЗАЦИИ ОТКРЫТЫХ КАРТ С ПОМОЩЬЮ ТЕХНОЛОГИИ КОМПЬЮТЕРНОГО ЗРЕНИЯ**

***Никита Вячеславович Порохов***

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

***Пётр Юрьевич Бугаков***

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

В условиях меняющегося современного мира, все больше карт нуждаются в постоянном обновлении. Бурно растущие города, изменение ландшафта, вследствие человеческой деятельности, все это сказывается на навигации на местности и достоверности данных, отображенных на карте. Чтобы исправить данную ситуацию, необходимо использовать, современные технологии, которые дают множество возможностей для обновления информации на картах. Каждый год появляется новая технология, в частности, большое распространение в последние время получила технология компьютерного зрения. Целью статьи является описание метода автоматизированной актуализации открытых карт, таких как OpenStreetMap, с использованием технологии компьютерного зрения. Для возможной апробации данной методики были представлены рекомендации по выбору и разработке программного обеспечения. Компьютерное зрение реализовывалось с помощью библиотеки OpenCV, а для разработки бизнес-логики проекта использовался язык программирования Java вместе с системой управления базами данных PostgreSQL.

**Ключевые слова:** автоматизация, открытые карты, компьютерное зрение, OpenCV, OpenStreetMap

## **ALGORITHM FOR DEVELOPING AUTOMATED UPDATING OF OPEN MAPS USING COMPUTER VISION TECHNOLOGY**

***Nikita V. Porokhov***

Siberian State University of Geosystems and Technologies, 10, Plakhotnogo St., Novosibirsk, 630108, Russia, Master student, 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

In a constantly changing world many maps need regular updating. Rapidly growing cities, changes of landscape affect the navigation on the ground and the reliability of data displayed in the map. To improve this situation, it is necessary to use modern technologies that provide many opportunities for updating information in maps. Every year a new technology appears, in particular, computer vision technology has become widespread in recent years. The purpose of the article is to develop and describe a method for automated updating of open maps, such as OpenStreetMap, using

computer vision technology. For possible testing of this technique, recommendations were presented on the selection and development of software. Computer vision was implemented with using the OpenCV library, and the Java programming language was used to develop the business logic of the project together with the PostgreSQL database management system.

**Keywords:** automation, open maps, computer vision, OpenCV, OpenStreetMap

## *Введение*

Одним из способов отображения геопространственной информации является электронная карта. Однако, обновление карты, которая поддерживается крупной компанией, может занимать достаточно большой промежуток времени. В свою очередь открытые карты, такие как OSM, дают возможность редактирования в реальном времени всем зарегистрированным пользователям, с последующим отображением изменений на карте. OpenStreetMap предоставляет простой интерфейс для редактирования данных карты, а также позволяет скачивать необходимый тайл карты в формате .osm или последнюю полную копию базы данных OpenStreetMap в формате XML.

Применение новых технологий для актуализации открытых карт позволяет:

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

Современные технологии уже повсеместно используются в похожих задачах [1–3]. Однако большинство из них используют для обновления карт спутниковые или аэрофотоснимки. Представленная методика обновления оперирует данными, полученными с помощью наземной съемки и специального оборудования.

## *Методы и материалы*

Для реализации технологии компьютерного зрения необходимо использовать программные библиотеки OpenCV и TensorFlow. OpenCV – это библиотека компьютерного зрения и машинного обучения с открытым исходным кодом. В нее входят более 2500 алгоритмов, в которых есть как классические, так и современные алгоритмы для компьютерного зрения и машинного обучения [4–6]. Эта библиотека имеет интерфейсы на различных языках, среди которых Python, Java, C++ и Matlab.

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 [7, 8]. Применяется как для исследований, так и для разработки собственных продуктов Google. Основной API для работы с библиотекой реали-

зован для Python, также существуют реализации для R, C Sharp, C++, Haskell, Java, Go и Swift.

Модуль по распознаванию здания будет написан на языке Python. Большинство библиотек пишутся именно под этот язык [9]. Это легкий высокоуровневый объектно-ориентированный язык программирования.

С помощью данных библиотек необходимо создать и обучить нейронную сеть, которая реализует технологию компьютерного зрения. Чтобы нейронная сеть была способна идентифицировать здание среди множества объектов улицы, необходимо ее обучить с помощью набора данных. Существует множество наборов данных, с помощью которых тренируется нейронная сеть [10, 11], однако, для российских городов наблюдается дефицит таких наборов. Для обучения был взят набор данных Sheffield building image dataset [12]. Набор включает в себя 3000 изображений зданий города Шеффилд. Как видно на рисунке 1 здания изображения зданий получены под разным углом и были отсняты в разное время и с разным освещением. Чтобы обеспечить более высокую эффективность вычислений и низкие требования к памяти изображения предоставлены в разрешении 160 на 120 пикселей [13].

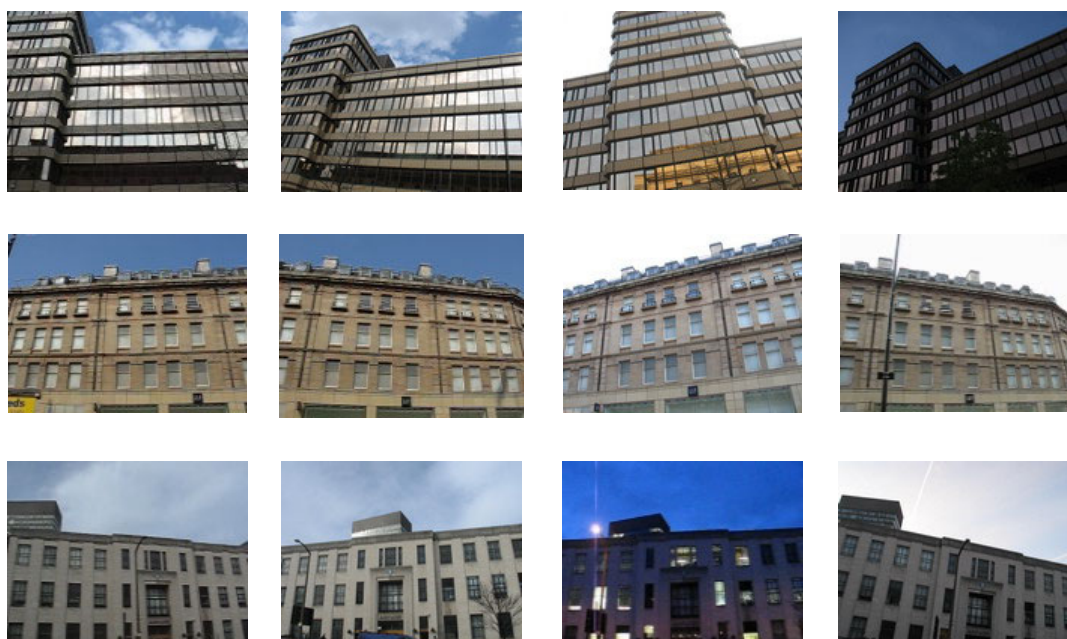


Рис. 1. Пример набора данных для обучения нейронной сети

После обучения нейронная сеть будет способна распознавать здания на изображениях.

### *Результаты*

Суть метода заключается в распознавании зданий на фотоснимках, полученных при наземной съемке улицы, с последующим вычислением координат зда-

ния и отображением его на карте OSM, если объект до этого не был нанесен на карту.

На рис. 2 представлена схема, показывающая этапы метода актуализации. На автомобиль устанавливается оборудование, состоящее из камер, GPS-приемника и дальномера. Автомобиль едет по заранее заданному маршруту, а камера ведет непрерывную съемку прилегающих к улице объектов инфраструктуры.

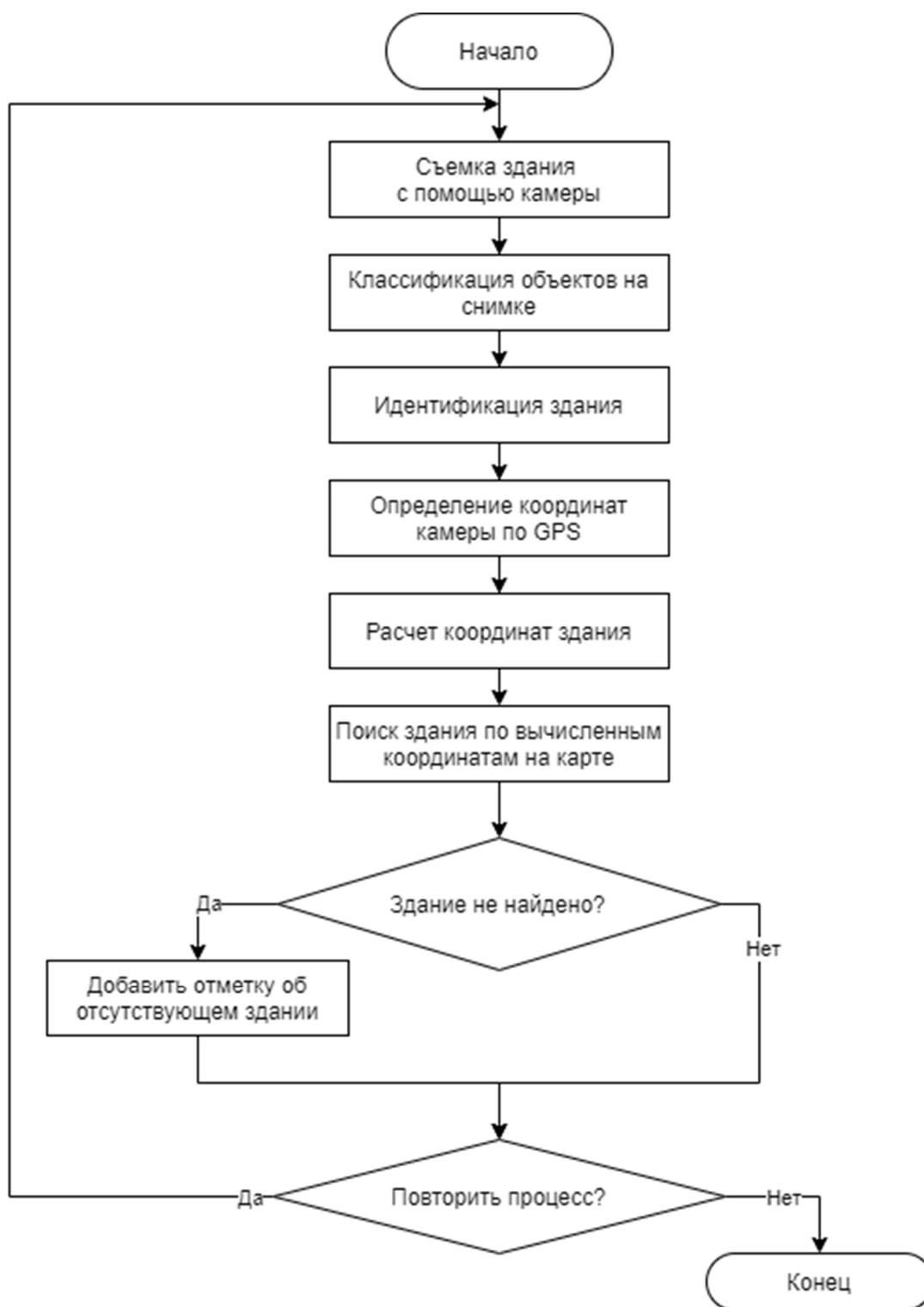


Рис. 2. Схема разрабатываемого метода актуализации

Данные с камеры поступают на вход нейронной сети, которая классифицирует все объекты на снимке или кадре. Это делается для того, чтобы идентифицировать здание из множества объектов, которые не нужны для дальнейшей обработки. Из множества объектов изображенных на снимке необходимо выбрать именно здание, для этого все объекты необходимо классифицировать. Для этого существуют следующие способы:

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 [14].

Полученные данные, в ходе распознавания, загружаются в базу данных, созданную в СУБД PostgreSQL. PostgreSQL – свободная объектно-реляционная система управления базами данных. Основным преимуществом является то, что она бесплатная [15, 16]. Также она зарекомендовала себя на рынке СУБД, как надежная и отказоустойчивая система.

В базе должны храниться следующие данные, полученные со съемки:

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

После добавления всех данных выполняется расчет координат зданий по координатам автомобиля и расстояния до здания.

На следующем этапе данные передаются на вход программному модулю, который определяет нанесен ли объект на карту. Для реализации модуля необходим доступ к данным OpenStreetMap, для сравнения координат. Чтобы определить, являются ли вычисленные координаты зданием, строится буферная зона вокруг данной точки. Для этого по маршруту следования автомобиля формируется тайл в OSM для скачивания данных (пример тайла приведен на рис. 3).

Когда тайл сформирован, происходит скачивание информации по всем объектам, попадающим на данный тайл. Данные скачиваются в формате osm, он не очень удобен для работы, поэтому нужно конвертировать в формат json или geojson. Данные, скаченные с OSM, имеют теги, наличие которых облегчает поиск необходимых объектов и их координат, в частности зданий (building). При конвертации данных в формат geojson, им присваиваются реальные координаты вместо ссылок на эти координаты, которые расположены в базе OSM [17].

Когда все данные подготовлены, вокруг исследуемой точки строится буферная зона [18]. В случае если, буферная зона перекрывает точку с известными координатами, помеченную тегом «building» или находится внутри полигона, построенного по скаченным точкам, то она пропускается. Это означает, что объект уже нанесен на карту и нет необходимости информировать 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.

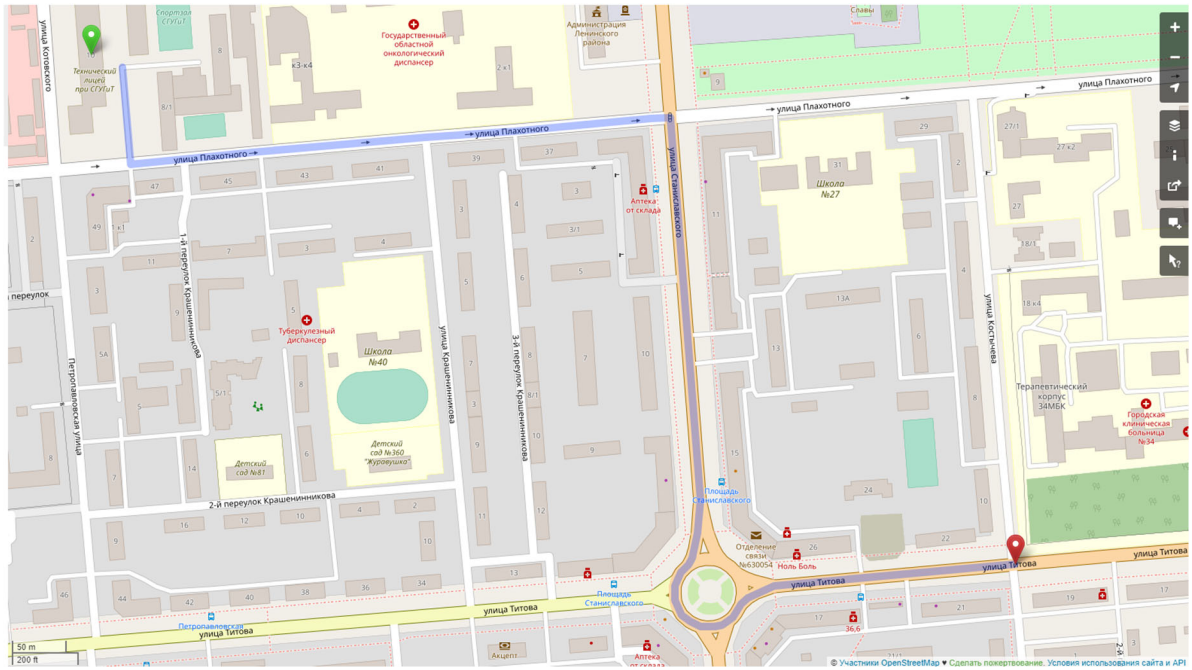


Рис. 3. Пример тайла карты с маршрутом

Данный модуль предлагается разработать на языке программирования Java, который представляет собой высокоуровневый объектно-ориентированный язык программирования [19, 20]. Данный выбор обусловлен тем, что Java является кроссплатформенным языком программирования [21], что обеспечивает легкость реализации web-взаимодействия с OSM. Также на Java написан один из редакторов OSM, что должно помочь при разработке.

### Заключение

Была описана методика автоматизированной актуализации открытых карт с помощью технологии компьютерного зрения. Предложен набор программных средств, с помощью которых в будущем планируется реализовать данную методику в готовую функционирующую систему. Перечень программных и аппаратных средств представлен в таблице.

#### Перечень программных и аппаратных средств

Средство	Назначение
Камера	Получение изображения, на котором расположено здание
OpenCV, TensorFlow	Библиотеки машинного обучения. Используются для классификации объектов, изображенных на снимке и идентификации здания
GPS-приемник	Отслеживание координат в каждый момент съемки
Python, Java	Языки программирования. На Python будет создана нейронная сеть для распознавания здания среди других объектов, изображенных на снимке. На Java будет написан модуль идентификации объекта на карте
PostgreSQL	СУБД для создания базы данных, в которой будут храниться данные о съемке

Для более точного определения зданий, типичных для Российской Федерации, планируется создать свой набор данных. Он будет состоять из большого количества снимков зданий с разных ракурсов и при разном освещении.

### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Бойко, И. А. Распознавание объектов на основе видеосигнала, полученного с камеры, установленной на подвижной платформе / И. А. Бойко, Р. А. Гурьянов. – Текст: непосредственный // Молодой ученый. – 2013. – № 6 (53). – С. 34-36. – URL: <https://moluch.ru/archive/53/7211/> (дата обращения: 10.12.2020).
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. OpenStreetMap Forum [Электронный ресурс]. – Режим доступа: <https://forum.openstreetmap.org/viewtopic.php?id=55220> (дата обращения: 05.12.2020).
4. Буэно, Суарес, Эспиноса. Learning Image Processing with OpenCV. – М.: ДМК-Пресс, 2016. – 210 с.
5. Кэлер А., Брэдки Г. Изучаем OpenCV 3. – М.: ДМК-Пресс, 2017. – 826 с.
6. Прохоренок Н. OpenCV и Java. Обработка изображений и компьютерное зрение. – СПб.: БХВ-Петербург, 2018. – 320 с.
7. Джулли А., Пал С. Библиотека Keras – инструмент глубокого обучения. Реализация нейронных сетей с помощью библиотек Theano и TensorFlow. – ДМК-Пресс, 2017. – 294 с.
8. TensorFlow [Электронный ресурс]. – Режим доступа: <https://www.tensorflow.org/> (дата обращения: 01.12.2020).
9. Коэльо Л. П., Ричерт В. Построение систем машинного обучения на языке Python. – Перевод с английского. – М.: ДМК Пресс, 2015. – 303 с.
10. J. Philbin, O. Chum, M. Isard, J. Sivic, A. Zisserman, Object retrieval with large vocabularies and fast spatial matching, in: IEEE Conf. Computer Vision and Pattern Recognition, 2007.
11. T.S.H. Shao, L.V. Gool, Zubud-Zurich buildings database for image based recognition, Technique Report No. 260, Swiss Federal Institute of Technology, 2003.
12. J. Li, N.M. Allinson, Subspace learning-based dimensionality reduction in building recognition, Neurocomputing 73 (1–3) (2009) 324–330.
13. J. Li, N.M. Allinson, Building recognition in urban environments: A survey of state-of-the-art and future challenges, Information Sciences Volume 277, 1 September 2014, Pages 406-420.
14. Зеленцов И.А. Распознавание образов. Обзорная лекция. URL: [http://it-claim.ru/Persons/Zelencov/Lecture\\_presentation.pdf](http://it-claim.ru/Persons/Zelencov/Lecture_presentation.pdf) (дата обращения 22.10.2020).
15. Иван Панченко. PostgreSQL: вчера, сегодня, завтра. Открытые системы. СУБД, № 03, 2015 (дата обращения 14.11.2020).
16. PostgreSQL [Электронный ресурс]. – Режим доступа: <https://www.postgresql.org/> (дата обращения: 03.12.2020).
17. GEOJSON [Электронный ресурс]. – Режим доступа: <https://geojson.org/> (дата обращения: 22.01.2021)
18. ArcMap [Электронный ресурс] – Режим доступа: <https://desktop.arcgis.com/ru/arcmap/> (дата обращения 10.02.2021).
19. Герберт Шилдт. Java. Полное руководство, 10-е издание. – М.: «Диалектика», 2018. – 1488 с.
20. Oracle Java [Электронный ресурс]. – Режим доступа: <https://www.oracle.com/ru/java/> (дата обращения: 07.12.2020).
21. Брюс Эккель. Философия Java. – 4-е изд. – СПб.: Питер, 2018. – 1168 с.

© Н. В. Порохов, П. Ю. Бугаков, 2021