Article

Towards Automatic Vandalism Detection in OpenStreetMap

Pascal Neis *, Marcus Goetz and Alexander Zipf

Geoinformatics Research Group, Department of Geography, University of Heidelberg,
Berliner Street 48, D-69120 Heidelberg, Germany; E-Mails: m.goetz@uni-heidelberg.de (M.G.);
zipf@uni-heidelberg.de (A.Z.)

* Author to whom correspondence should be addressed; E-Mail: neis@uni-heidelberg.de;
Tel.: +49-6221-54-5504; Fax: +49-6221-54-4529.

Abstract: The OpenStreetMap (OSM) project, a well-known source of freely available worldwide geodata collected by volunteers, has experienced a consistent increase in popularity in recent years. One of the main caveats that is closely related to this popularity increase is different types of vandalism that occur in the projects database. Since the applicability and reliability of crowd-sourced geodata, as well as the success of the whole community, are heavily affected by such cases of vandalism, it is essential to counteract those occurrences. The question, however, is: How can the OSM project protect itself against data vandalism? To be able to give a sophisticated answer to this question, different cases of vandalism in the OSM project have been analyzed in detail. Furthermore, the current OSM database and its contributions have been investigated by applying a variety of tests based on other Web 2.0 vandalism detection tools. The results gathered from these prior steps were used to develop a rule-based system for the automated detection of vandalism in OSM. The developed prototype provides useful information about the vandalism types and their impact on the OSM project data.

1. Introduction

User generated content (UGC) [1] is a well-known phenomenon of the Web 2.0 movement. With the ubiquitous availability of GPS-enabled devices, such as smartphones, cameras or tablets, users not only collected UGC, but also started to geo-reference their collected information. This new trend became popular under the term "Volunteered Geographic Information" (VGI) [2], or crowd-sourced geodata [3]. Amateurs and professionals collaboratively collect, share and enhance geodata for specific VGI platforms. Essentially, everybody is able and allowed to use available VGI for their own applications and services at no charge.

One of the most popular and most manifold projects for VGI is the OpenStreetMap (OSM) project [4–6]. Initially aiming at the creation of a global web map, the project soon turned far beyond that. Nowadays, OSM can be considered as a global geodata database that everybody can access, edit and use. A rapidly growing community and the emerging demand for open geodata made OSM one of the most used non-proprietary online maps and a source of geodata and information for many third-party applications, such as route planning & geocoding [7], 3D [8] or indoor [5] applications. Although users of OSM have to register prior to contributing, the OSM model has been designed under an open-access approach. However, although this open approach can be considered as the key to OSM's success, it can also be a source of a variety of problems. While most contributions are legitimate, some attacks by lobbyists or spammers result in vandalism. One of the most popular examples for vandalism in OSM is the case of two employees of a popular search engine deleting OSM features in the London area [9]. However, there has not yet been a general investigation on the impact and quantity of vandalism in OSM, and no distinct task force against vandalism has been implemented so far in OSM (*cf.* Wikipedia's "*Subtle Vandalism Taskforce*" [10]). The OSM "Data Working Group" can be contacted when "serious Disputes and Vandalism" [11] are encountered. However, "Minor incidents of vandalism should be dealt with by the local community" [11].

Nevertheless, following the idea of collective knowledge, it is assumed that vandalism is detected and corrected by other OSM contributors within an (unknown) period of time. In the case of OSM, vandalism can occur intentional and unintentional, contradicting the traditional definition of the term "vandalism". However, this strongly depends on the change itself (vandalizing a geometry is probably more obvious than vandalizing semantic information), as well as on the area in which the edit has been performed, e.g., vandalism in a metropolitan area will probably be detected and corrected pretty fast, whereas vandalism in a very rural area will potentially remain for a very long period of time.

In general, data validation and vandalism detection needs to be distinguished from each other. While data validation incorporates different methodologies for quality assurance, vandalism focuses on active data corruptions. In this paper, we will focus on the latter one. Although automatic methods for the detection of vandalism in OSM ought to be of interest for both the contributors as well as the consumers of the project, only a few tools or other significant developments have been accomplished in this field in recent years. At the time of writing, only two tools are available that can observe a predefined area of interest. The tools are "OpenWatchList" [12], which was developed under open source standards, and "OSM Mapper" developed by ITO [13]. Both tools provide the possibility to register to an RSS feed, which provides information about the latest changes in a distinct area. The functionality of the tools can be compared to so-called watchlists in Wikipedia, which track edits that

were made to selected articles [14]. There are many other similarities between OSM and Wikis, which nowadays are widespread on the Internet, such as the reliance on "radical trust" [15] and citizens that become the "sources of data" [14]. What all these tools have in common is that they inform registered users about every single edit in a predefined area, rather than evaluating the corresponding edit and highlighting potential cases of vandalism. That is, the user still has to (manually) investigate every single edit, which is a very time-consuming and tedious effort. Nevertheless, vandalism detection in OSM plays an important role and will gain more importance because of the increasing popularity of the project, which depends highly on its data quality [4,16].

Regarding Wikipedia, there are a couple of approaches available that focus on vandalism (*cf.* [17–20]). As mentioned by van den Berg *et al.* [14], it could be useful to apply similar approaches, methodologies and technologies, which have already been utilized in other open source projects and Web 2.0 encyclopedias, to detect vandalism in OSM and/or revert unconstructive changes.

Therefore, the main contribution of this paper is an investigation of vandalism in OSM, as well as the development of a rule-based system for the automated detection of vandalism in OSM. A comprehensive set of rules has been defined by investigating past vandalism incidents, the current OSM database and its contributions, as well as related Wikipedia vandalism detection tools. Our system incorporates the user's individual reputation, as well as the performed action. Both parameters are evaluated independent from each other, which allows an individual definition of weights for both parameters after the corresponding detection. Essentially, this enables dedicated OSM members (patrols) to individually define appropriate weights, as individual patrols probably judge the importance of a user's reputation differently. Obviously, the change itself is a very important component, which needs to be investigated. Therefore, the change is tested against well-known community criteria, such as provided attributes, or community accepted map features and more. Incorporating the user's reputation is another important factor, as investigations on vandalism for other Web 2.0 projects (*i.e.*, Wikipedia) revealed that in most cases, vandalism is performed by new (probably inexperienced) community members instead of more experienced community members [21]. The developed system architecture has been prototypically implemented and added to a minutely updated OSM database. In this way, it was possible to apply some initial early results to refine the defined base rules and further improve the automatic vandalism detection.

The remainder of this article is structured as follows: The following section gives a general introduction to the OSM project, while the third section of the paper describes the different types of vandalism that can occur in OSM. Sections four and five describe our developed rules-based system and the results that were gathered after applying our methodology. The last section discusses our findings and presents some suggestions for future work and research.

2. OpenStreetMap

Founded in 2004 at the University College London, the OSM project's goal is to create a free database with geographic information for the entire world. A large variety of different types of spatial data and features, such as roads, buildings, land use areas or Points-of-Interest (POI), are collected in the database. Following the Web 2.0 approach of a collaborative creation of massive data, any user can start contributing to the project after a short online registration. Essentially, every registered user is

able to add new elements, as well as alter or delete existing ones. This simple approach—similar to Wikipedia—led to more than 700,000 registered members by August 2012 [22]. In total, almost 200,000 members made at least one edit, and roughly 3% of all members made at least one change per month to the database by the end of 2011 [23], *i.e.*, those that can be considered as regular contributors.

Several different research studies about the quality and completeness of OSM data in comparison to other data sources (e.g., governmental or commercial) have been published in recent years (e.g., [4,24–27]). In Europe, OSM shows an adequate level of data coverage for urban areas, which allowed the development and distribution of map services or other applications, such as Location Based Serviced (LBS). However, less populated areas don't show the same completeness level in OSM, which makes the dataset unreliable in those areas. Thus, OSM data can be very use-case dependent, and the requirements must be carefully considered [28].

As stated above, every community member can alter the current OSM database; however anonymous changes are not supported [4]. This is a crucial difference between Wikipedia and OSM and brings at least a small advantage: "OSM users are identified by their usernames, which can be blocked. In Wikipedia, users are identified by username or IP-address and more than one user might use the same IP-address" [14]. However, after uploading the data to the community, the change is instantly live, *i.e.*, applied to the productive system. Essentially, there is no quality or vandalism control prior to the publication, in contrast to, for example, Google MapMaker [29], where experienced users review new submitted content. General thoughts on how an OSM user should react in the case he or she detects vandalized data in the project can be found on the vandalism webpage in the OSM Wiki [30].

Registered users can contribute data to OSM in different ways. The classic approach is to collect data with a GPS receiver, which afterwards can be edited with one of the various freely available editors, such as Potlatch or JOSM. Since November 2010, users are explicitly allowed to trace data from Bing aerial imagery and add the data to OSM [31]. This allows a user to collect information without physically being at a distinct location (thus a user from Germany can also provide data about a city in France). Regardless of how the data is collected, users can provide additional information, such as street names or building types, about the different OSM features. In the eight years since its initiation, 1.5 billion geo-referenced points (nodes in OSM terminology), 144 million ways (both linestrings and simple polygons) and 1.5 million relations (for describing relationships, such as turn restrictions or complex polygons with holes) [32] have been collected as of today (August 2012).

Each of the objects contains a version number, a unique ID, the name of the last editor and the date of the last modification (*i.e.*, the date of creation for new objects). Furthermore, so-called tag-value pairs containing additional (user-provided) information are attached to each feature. Any modification made by a user to a feature in OSM is stored in a so-called changeset, containing information about the change itself, as well as the editor and the time of the edit.

If a user wishes to implement the data of the OSM project for an application (for example a map for public transportation), a planet file, which contains all information of the latest database of the project, can be downloaded [33]. However, as people are contributing data every minute, a dataset will become outdated after a short period of time (probably even after one minute). To avoid the deployment of a full OSM database every time an update is required (which is hardly feasible for a minutely or hourly

updated database), the OSM project provides so-called OSM Change-Files (OSC), also referred to as "Diff", which can be downloaded. These files only contain the latest changes to the database and are available for different time frames, such as every minute, hour or day. The format of the OSC-files will be explained in more detail in a later section of this paper. Figure 1 shows a simplified version of the OSM project infrastructure. By using one of the freely available OSM editors, the contributors can edit any object of the project's database. External applications, such as routing or mapping applications, can use the project's data by retrieving the dump- and diff-files from the database.

Figure 1. The OpenStreetMap Infrastructure/Geostack (simplified).
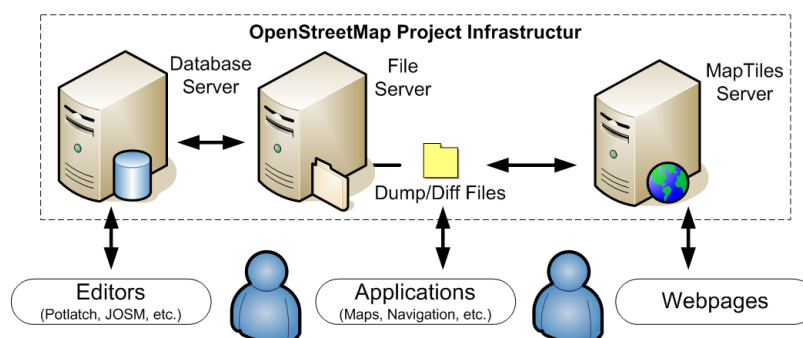
On average, nearly 700 new members have registered to the project each day between January and March 2012. According to [23], nearly 30% of those newly registered contributors will become active contributors (and not only a registered user). That is, each day in 2012, 230 new OSM members started contributing to OSM. Table 1 contains the average number of edits per OSM object (node, way and relation) per day between January and June 2012.

Table 1. Number of daily edited OSM objects (January–June 2012).

Number of ...	Node	Way	Relation
Daily created objects	1,200,000	130,000	1,500
Daily modified objects	170,000	70,000	3,500
Daily deleted objects	195,000	16,000	280
Users who daily edited	2,000	1,940	560

Considering these numbers for estimating future processing workloads for our suggested tool, it can be determined that (on average) every minute, 830 node creations, 190 node modifications and 135 node deletions will have to be performed. Furthermore, 90 new, 48 modified and 11 deleted ways and one new, two modified and 0.2 deleted relations have to be processed. Those numbers can obviously vary during the day, but they give a first indication on how much data will be edited.

3. Types of Vandalism

The open approach of data collection in the OSM project can cause a variety of types of vandalism. It is possible that a contributor purposely or accidently makes changes to the dataset that are harming the project's main goal. Common vandalism types that appear in the actual OSM geodata database are (based upon [30]):

- a new object with no commonly used attributes
- a non-regular geometrical modification of an object ("Graffiti")
- a non-common modification of the attributes of an object
- randomly deleting existing objects
- an overall abnormal behavior by a contributor
- generating fictional and non-existing objects
- inappropriate use of automated edits (bots) in the database
- application of mechanic edits (e.g., selecting 100 buildings and adding the key building: roof: shape = flat)

Other vandalism types can be found in the OSM project, but that are not solely limited to the actual geodata that is stored in the database are:

- Copyright infringements, e.g., tracing data from Google Maps
- Disputes on the project-wiki
- Disruptive behavior or spamming of a member (e.g., in his edits, in the forum, on the mailing lists or on her/his user page)

The following Figure 2 shows an example of "Graffiti" vandalism in 2011 in Zwijndrecht (The Netherlands). The users who caused this disorder of the features used the Potlatch OSM editor, which directly applies the changes to the live OSM database.

Figure 2. Example of "Graffiti" vandalism in OSM in Zwijndrecht (The Netherlands) [34].
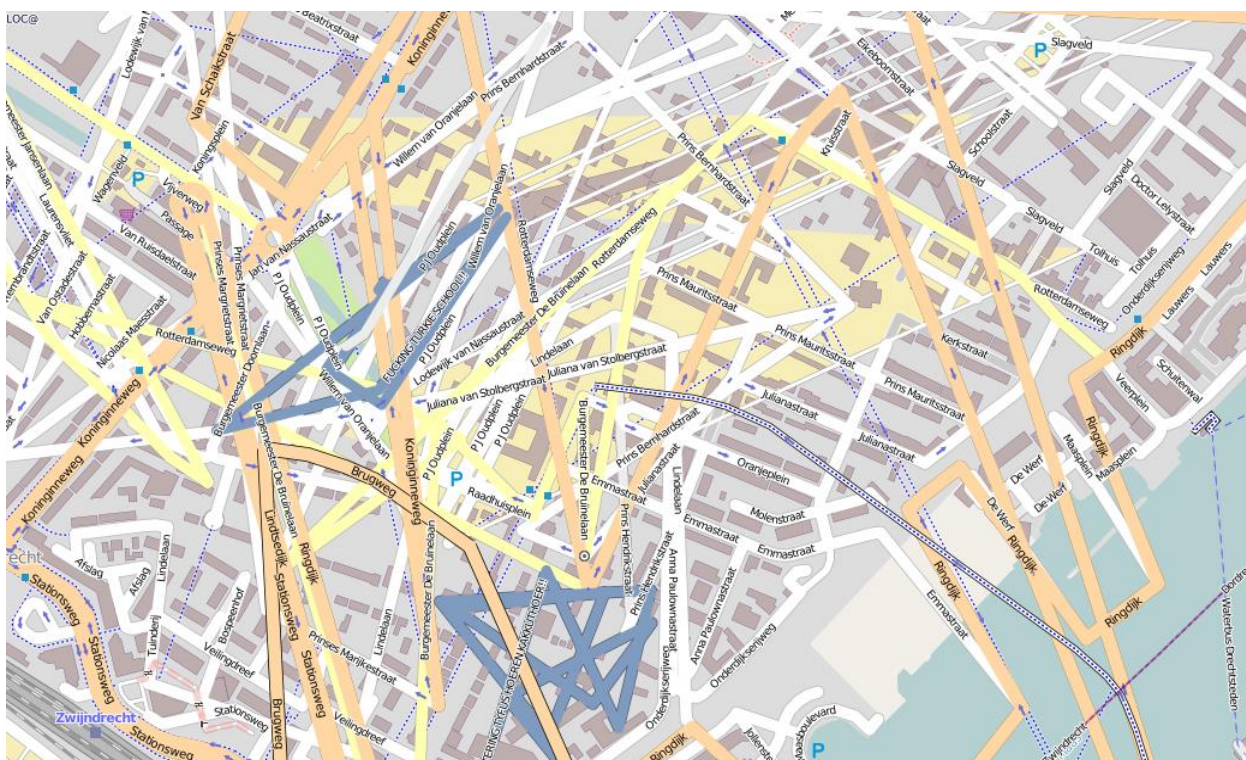
Potthast *et al.* [35] and West *et al.* [21] manually analyzed vandalism in Wikipedia to learn about the specific characteristics. For our analysis, we used a similar approach and manually analyzed 204 user blocks of the project to gather more detailed information about vandalism in OSM. Members

of the OSM Data Working Group or moderators are allowed to block other OSM members for a short period of time (between 0 and 96 h) [36]. The blocked members need to login to the main OSM website and read their notification to be able to unblock their accounts. There are several reasons for such a block, such as importing data that infringes the Import-Guidelines [37], creating mass-edits which do not follow the Code-of-Conduct [38] or vandalizing the data in some sort of way. Out of these 204 users that were blocked between 7 October 2009 and 31 July 2012, we were able to determine 51 cases of data vandalism events, not counting members that were blocked multiple times. The geographical pattern showed that cases of vandalism can be found worldwide in the OSM database with a slight focus on larger cities. Table 2 summarizes our results of manually collected vandalism cases and their characteristics.

Table 2. Characteristics of Vandalism in OSM (October 2009–July 2012).

Feature	Value	Description
Fictional Data	33.3%	The user created some fictional data
Editing Data	33.3%	The user modified some existing data, e.g., did some non-regular geometrical modification
Deleting Data	43.1%	The user deleted some existing data
New User	76.4%	The data was vandalized by a new project member
Potlatch Editor	82.4%	OSM editor which was used during the vandalism

When a vandalism event on one of the articles in Wikipedia is detected, it is usually reverted within a matter of minutes [39]. In our OSM analysis, 63% of the vandalism events were reverted within 24 h and 76.5% within 48 h. Some outliers could be determined, which needed more than 5 d up to a maximum of 29 d.

As mentioned before, there are three types of objects used in the database: nodes, ways and relations. All three object types can be created, modified or deleted by each member of the project. Essentially, each member can also alter objects that have been created or altered previously by a different community member. Changes to the OSM database can be analyzed by investigating the "Diff"-files (also termed OSM-Change files). Every "Diff"-file, whether it contains information about each minute, hour or day, provides information about all changes that have been made to the database within this distinct time frame. The most recent edits, that is, those changes that happened after the creation of the previous diff file, are grouped according to their action ("create", "modify" or "delete") and object type ("node", "way" or "relation"). However, the diff-files only contain the geo-reference of the nodes (*i.e.*, longitude and latitude), but not the actual geometry of way- or relation-features. Additionally, if a feature has been modified, the file does not provide any specific information about the actual change. If, for example, a node has been moved, the change file will only contain the new location of the node and not both locations (or the difference vector). The action types "create" and "delete" are only valid for the creation or deletion of a geometric object (e.g., the creation of a node or the deletion of a complete relation). In contrast, whenever additional (semantic) information is added (*i.e.*, create), altered (*i.e.*, modify) or removed (*i.e.*, delete), this change is represented as a modify change, rather than a create or delete action. For a more accurate analysis of the object (especially for a

modify action), it is therefore often necessary to gather the history of the object. If the change constitutes the creation of an object, it is necessary to have a closer look at what type of object has been created and, for example, to check which attributes have been used for the initial creation. In the case of a modification of an existing object, additional aspects need to be considered, such as: Has the geometry or the attributes of the object changed? Being able to provide information about these particular changes, the former and the updated object need to be compared. Additional useful information can be gathered by determining the former object owner, the version number and the date of the former object. If the change that has been made to the database is a deletion of an OSM object, similar characteristics of the modification analysis process, such as determining the type of object and what the prior object metadata looked like, should be considered.

4. Rule-Based Vandalism Detection System

The detection of the introduced types of vandalism in the database allowed for the prototypical implementation of a rule-based decision system, named *OSMPatrol*. One of the main goals of the prototype was to detect the vandalism as fast as possible. For this purpose we used the OSM "Diff"-files and the contained information about changes to the database that are made each minute. Additionally, an OSM database was created to be able to compare the former and the new OSM object. As described, users in OSM can basically provide any kind of additional (semantic) information by tagging the corresponding OSM feature with key-value pairs. Both, the key and the value can typically contain any arbitrary content. Nevertheless, there are community-accepted and well-known OSM Map Features [40]. To be able to compare and judge the vandalism-likelihood of the additional (semantic) information, our prototypical application matches the added (or altered) information against the well-known OSM Map Features. This is achieved by parsing the OSM wiki page for the map features, extracting the different features (*i.e.*, key-value pairs) and storing them in a database. It was decided to use both the English and the German version of the website as a reference, because these typically contain the most details. As of 13 August 2012, there was a total of 1,139 map features in our database. When evaluating the individual changes, the applied key-value pair is tested against the database with all the map features. Additionally, to retrieve more information about the OSM user, we built a similar database as implemented by Neis [41]. This contributor table contains detailed information such as:

- How many nodes, ways and relations an OSM contributor creates, modifies or deletes?
- What is her/his date of registration? When did she/he start to contribute?
- How often did she/he use one of the most common Tags on an OSM object such as: address, amenity, boundary, building, highway, landuse, leisure, name, natural, railway, sport or waterway?

Both tables are stored in the *OSMPatrol* PostgreSQL database. Figure 3 shows the complete architecture of the developed prototype in relation to the OSM project architecture. To retrieve the OSM "Diff"-files, which contain the changes that were made to the database per minute, the OSMOSIS tool is applied. OSMOSIS [42] is an open-source command line JAVA tool, which processes OSM data in several different ways. In our particular case, it was also important to update

the newly created OSM PostgreSQL database with the "Diff"-file information to be able to compare the former and newly created or updated OSM objects.

Figure 3. OSM & *OSMPatrol* Architecture.
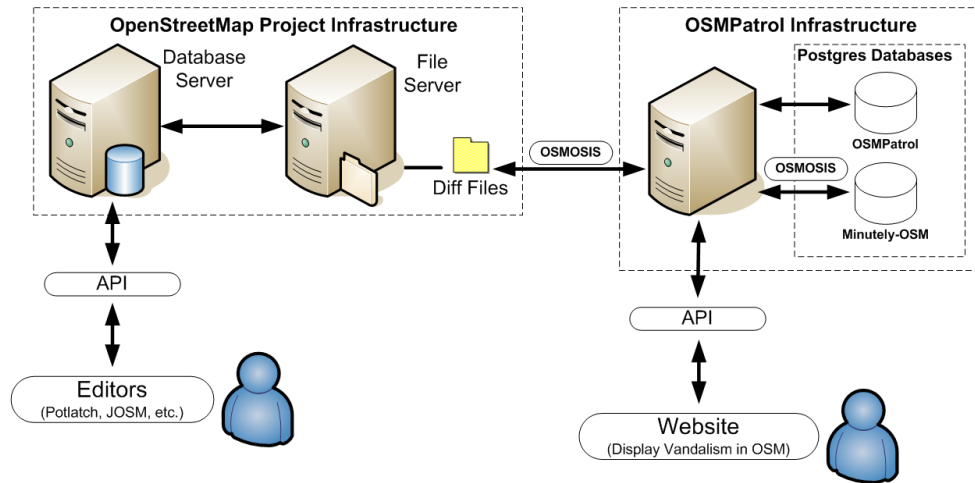
For a better explanation of the different processing steps that are executed every minute to detect potential vandalism in the database, Figure 4 shows a UML sequence diagram.

Figure 4. UML sequence diagram of the vandalism detection tool (*OSMPatrol*).
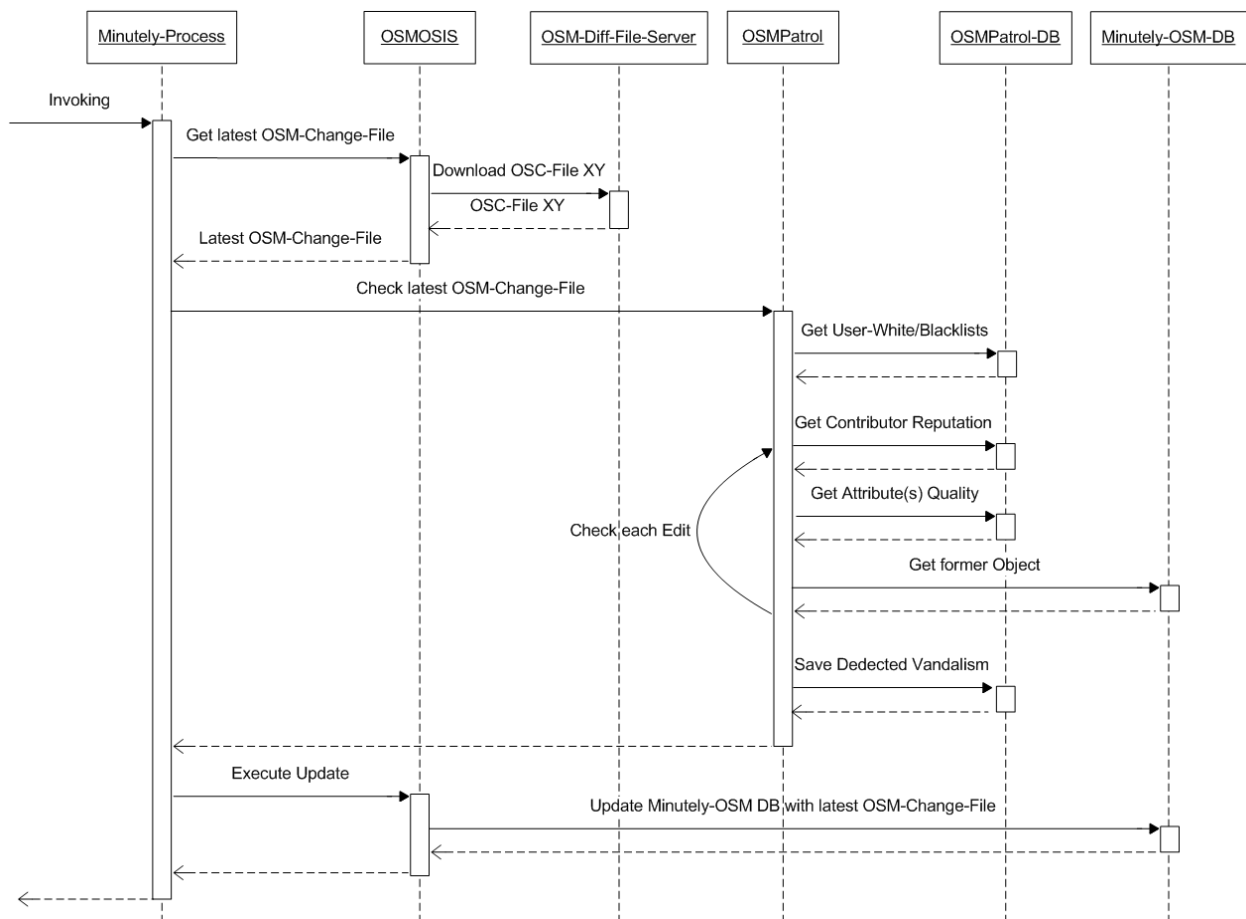
The process starts with the download of the latest OSM "Diff"-file via OSMOSIS. As soon as the download has finished, the main tool, *OSMPatrol*, analyzes this file for signs of vandalism. Prior to testing each edit of the retrieved file, the tool requests two lists that we defined and contain the following information: (a) a list of users who have a history of vandalism incidents (black-list); and, (b) a list of users who should not be considered during the test for vandalism (white-list). During the reviewing process of each OSM edit of the "Diff"-file, information, such as the contributor reputation, the quality of the attributes that have been used and, if necessary, the former object versions, are requested. If an edit is detected as vandalism, it will be stored in an extra table with some additional information. After testing all edits, the OSMOSIS tool will update the OSM database with the regular "Diff"-file, which is an important last step.

Besides the regular architecture and the sequence of the vandalism tool, a major aspect is the assignment of each edit to the corresponding vandalism type. As described in the diagram (Figure 4), every edit in OSM will be tested. One assumption that can be made is that new contributors that just joined the project are more prone to errors, mistakes or vandalism in comparison to a more experienced member. Thus, the overall value, which describes if an edit is a potential act of vandalism, is separated into two parts: the first value summarizes the contributor reputation and the second value rates the edit itself. Additionally, it is possible to filter the results by the corresponding vandalism type and/or by the edits that were created by new contributors. However, to be able to determine a value that represents the reputation of a member, several values need to be integrated. The created objects and corresponding tags that were used during the creation should be equally considered in the reputation value of a user. The aforementioned contributors of the project create more nodes/ways, thus the relation object gets a smaller weight. The used Tags are divided into the "Top12" used Tags of the project. Those contain all established key categories of the OSM Map Features [40] list. The reputation value for a contributor can be between 0 and 100%, where 0 represents the value of a new member and 100% an expert member. The following list shows the weight of each aspect to calculate the final user reputation:

- A maximum of 20% for the number of created nodes by the contributor
- A maximum of 20% for the number of created ways by the contributor
- A maximum of 12% for the number of created relations by the contributor
- A maximum of 4% for the number of each of the "Top12" Tags (address, amenity, boundary, building, highway, landuse, leisure, name, natural, railway, sport or waterway)

The registration date of a member was not taken into account for determining the reputation of a contributor, but it was used during the vandalism detection process. Figure 5 shows a UML activity diagram for the process of potential vandalism detection of an OSM edit. In general, it can be distinguished between the creation, modification and deletion of an object. In a first step, the contributor who edited the object will be determined. Depending on what type of edit has been made, a value above 0% determines if an edit can be considered as vandalism or not.

If the edit is a newly created object, only the attributes of the object, if available, will be used for designation. At the same time, all attributes will be matched against the map features list (*cf.* above). If a combination is not available in the map feature list, the vandalism-value of this edit is increased. Thus, the overall vandalism value of an edit can increase according to its number of tags.

If the edit is a modification or deletion of an object, some additional parameters will be checked. The last version of the object is compared with the edit (*i.e.*, the new version), to get answers to questions such as: Who is the former object-owner and does he show a high user reputation value? What is the former object version number, and what is the edit date of the former object? All three items will be used again to increase the overall vandalism value of an edit. Additionally, if the edit is a modification, the geometries will be compared with each other to detect if an object has been moved more than, e.g., 11 m. Also the tags of the latest and the former object will be compared to analyze which tag (key/value) has been changed, added or removed.

Figure 5. UML activity diagram to detect the types of vandalism of an OSM edit sequence diagram of the vandalism detection tool (*OSMPatrol*).
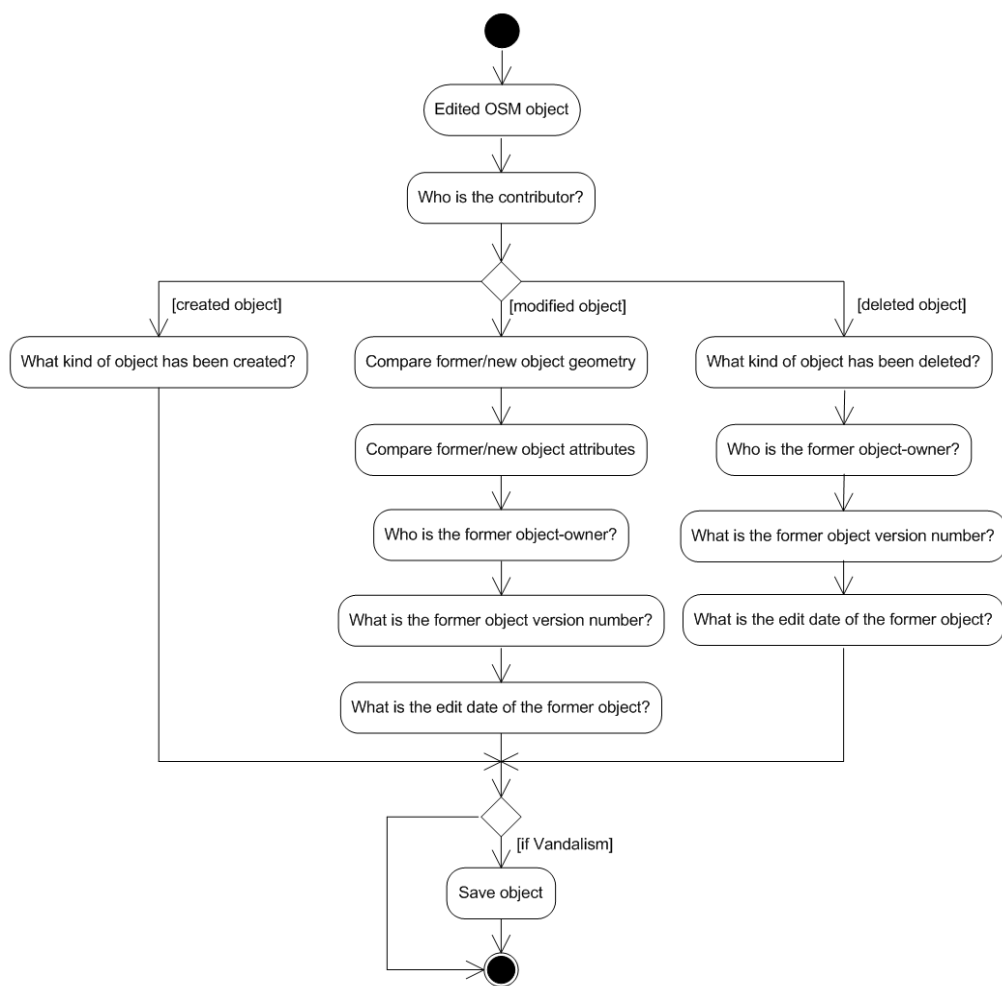

5. Experimental Results

After testing the developed prototype for small areas with known heavy and light vandalism cases, we conducted our final analysis by running the prototype on a dedicated server for one week (14 August 2012–21 August 2012). The server's hardware consisted of a 16 core CPU with 2.93 GHz, 35 GB of RAM and overall 3 TB hard disk space with speeds between 5,400 and 7,200 RPM. During the testing phase, *OSMPatrol* detected about seven Mio "vandalism" edits of 9,200 different users for the entire week. During the same time frame, around 16 Mio edits were made to the OSM database.

This means that the prototype marked 44% of all edits as possible vandalism. The following Figure 6 shows the distribution of the affected amounts of nodes, ways and relations. Additionally, the figure provides information about how many of the affected objects were detected during a creation, modification or deletion.

Figure 6. Distribution of objects and edit-types in the detected vandalism (14–21 August 2012).
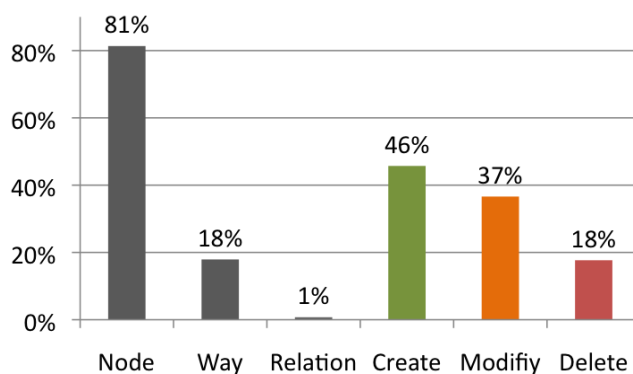
As described by Neis and Zipf [23], this week basically represents an average week (regarding contribution behavior), meaning that the OSM members contribute to the project in a similar way every other week. A similar statement can be made about the vandalism edits. We were not able to determine a particular day of the week on which a higher number of vandalism edits took place. Interestingly, almost 1/3 of the 9,200 users who were detected as possible vandalism committers were new contributors to the project. The following Figure 7 shows the distribution of edits that were detected as vandalism based on the user reputation. About 50% of the users, for which *OSMPatrol* detected a possible case of vandalism, have a user reputation larger than 66%, indicating that also experienced contributors' actions could be recognized as vandalism. Based on the collected results, users with a reputation level larger than 66% committed 48% of the detected possible vandalism cases. According to these values, about 43% of all detected vandalism edits were committed by new users of the project with a low reputation. Overall, almost 1/3 (36%) of all vandalism users were new users.

Figure 7. Distribution of vandalism users and vandalism edits based on the user reputation (14–21 August 2012).
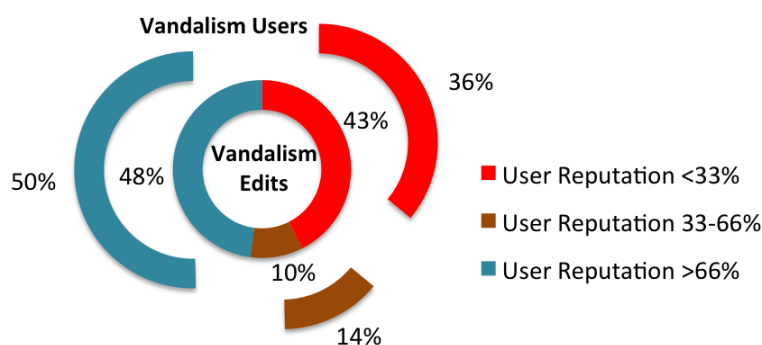
Different methods were applied to detect potential cases of vandalism. For each saved vandalism edit there are additional attributes available, e.g., timestamp, user and her/his reputation, vandalism value and a text comment with some information why the edit is marked as vandalism.

Based on these values we created three basic filters:

1. Show all edits of new users and/or users with a very low reputation (<5%).
2. Show all users who modified or deleted more than 500 objects within one hour.
3. Show all users who modified node objects and moved the object for more than 500 m.

Based on the first filter, the edits of almost 500 users had to be checked every day. Filter two shows almost 75 users and filter three around 100 users per day in our test phase in August 2012. After investigating at least 30% of the edit provided by these users, it was possible to find at least one real vandalism case per day without taking a deeper look into the type of user edits. Every second case of these vandalism cases was reverted by other OSM contributors within one or two days. In some other cases, the users were blocked by the OSM DWG or the users were contacted via email to raise awareness about their editing errors to the live database. Generally, the results showed that it is feasible to detect real vandalism cases from our detected dataset. However, the analysis also shows that more tools are needed that support the user in analyzing the potential erroneous edit in OSM in an easier and more convenient way. One solution would be to provide a webpage or application that provides detailed information about the tag and/or geometrical changes of the history of an object, as introduced by Huggle [43] and his analysis of Wikipedia articles. With such an application or service, it would be easier to validate the vandalism edits that were detected by *OSMPatrol*.

Within the tested week, about 85% of the detected vandalism edits were committed by only 1,000 users. This shows that the importance of using and maintaining our introduced users black and white list cannot be overestimated. A few users were detected with cases of vandalism based on a large number of objects that were deleted by these users, which were created by the same user in the past. These special cases may allow for future research on how to separate edits made on the user's own data or data contributed by others. *OSMPatrol* was also able to detect a lot of deletions in France, where many active OSM users are currently cleaning up the prior import of buildings to the database.

Overall, *OSMPatrol* was able to detect vandalism types committed by new users, "illegal" imports and mass edits. However, it can be difficult to distinguish false positive vandalism types from actual cases committed by users with a high reputation level or by users who only delete one or two objects.

6. Discussion

When designing the rule base and actually implementing the prototype, a couple of issues and ideas for a more sophisticated prototype became apparent. Those have not been implemented yet, because incorporating those would come at the cost of not being able to perform minute vandalism detection (due to high computation costs). Nevertheless, they will be discussed in this section and maybe incorporated into the system later.

Regarding the individual user's reputation, it was questionable if (and how) to incorporate the project-membership time span (*i.e.*, the time since a user has registered). Is this parameter an indicator for vandalism or not? Is a change by a user who has been registered for four years probably less

vandalism than a change by a user who has registered one week ago? The pure incorporation of the project-belonging does not provide any new knowledge, because quite often users register without actually editing the database. However, after a couple of months (or years) they might come back and perform their very first edit—does this now mean that it is vandalism or not? Therefore, a combined consideration of the project-belonging and the activity in the community (a so-called activity-ratio) could lead to an additional indicator for the detection of vandalism; however a real implementation of this factor has not been realized in the presented prototype here.

As described above, massive changes on the geometry of a feature, such as moving a POI ten or more meters, is likely to be vandalism. However, being aware of this as a vicious contributor, it is possible to split the one geometric change into several small ones, which will probably not be detected. For example, instead of moving a POI 15 m at once to a different location, a user could also move the POI 15 times one meter to a different location. The former incident is detected as vandalism, whereas the latter one would not be detected, thus it is defined as being a safe change. Therefore, an extended and more sophisticated vandalism detector also needs to consider multiple changes for the same object over time. These might be detectable through time stamps that are close to each other for the same object by the same user.

Another questionable indicator for vandalism is the evaluation of the version number of an OSM feature. When creating a new feature, the version number is set to one. The version is incremented with each change of this distinct object (regardless of the actual change). But, is a change on an object with a high version number more likely a type of vandalism than a change on an object with a low version number? What about the opposite situation? A couple of investigations revealed that there are so-called "heavily edited objects" [44] in OSM, but it is not known if changes on those objects are more likely vandalism or not. One could argue that the higher the version number is, the less likely the feature is prone to vandalism, because a larger version value also means more potential feature reviewers. But what about one single user who changes an object a couple of times? Does this also indicate the correctness of the object? These factors indicate that investigations that solely rely on the version numbers are not a good indicator for the vandalism likelihood of an object. Nevertheless, combining the version number and the amount of distinct editors of an object probably represents an appropriate indicator.

When defining the rule base and designing the prototype, it also became apparent that vandalism detection is closely related to data validation (to some extent). One example for a potential indicator for the vandalism likelihood of a change in OSM is the consideration of the neighborhood or surrounding of a newly created or edited object. For example, changing a road in a residential area from residential to primary is very likely vandalism (or a necessary correction of a previous mistake by an inexperienced user). When only considering the change itself, these types of edits will not be detected, whereas the incorporation of the objects in the neighborhood probably provides additional justification for the detector. The fact that the neighborhood needs to be regarded is furthermore underpinned by Tobler's first law of geography [45], stating that, "Everything is related to everything else, but near things are more related than distant things".

Changing the name of an existing object or adding a name to a new object might also contain vandalism; however it is pretty hard (or potentially impossible) to properly distinguish between

vandalism and validation of names. For example, extending the abbreviation of a name to its full name (e.g., changing str to street) is not vandalism. As a possible solution, comparing names to an existing dictionary of common terms might provide clarity.

In contrast to the aforementioned issues, some vandalism related aspects, such as the IP address, cannot (yet) be implemented in the client due to missing data (it is not possible to gather the IP address of a OSM contributor). However, having such information might be a good (additional) indicator for the vandalism likelihood. It can be investigated if the IP (and the access point) suits the area in which the change has been performed. For example, if a user changes a street in a country that is hundreds of kilometers away, this might be more likely vandalism than changes of a street name next to the access point of a user, similar to what is described by West *et al.* [21] for Wikipedia vandalism detection. Additionally it could be useful to save the IP address of a user that commits the vandalism to block the user from the project and prevent any future vandalism.

7. Conclusions and Future Work

In this paper, we investigated past vandalism incidents, the current OSM database and its contributions, as well as related Wikipedia vandalism detection tools. Based on the results gathered, we developed a comprehensive rule-base prototypical tool that allows automatic vandalism detection in OSM. It can be concluded that it is (to some extent) possible to detect vandalism by applying a rule-based methodology. During the testing phase in August 2012, the prototype marked around seven million edits as potential vandalism. By creating several filters, we were able to determine at least one real vandalism case per day. Overall, the detected vandalism was committed by all types of OSM users and not only by new users or users with a low reputation.

However, as discussed in the previous section, there are some limitations and restrictions. Although those can be solved, they will likely come at the cost of slow performance, thus the initial aim of a minute detection can probably not be realized (at least in the presented prototype and the current server configuration). Furthermore, although vandalism can be detected, it needs to be stated that a manual review of the correctness is still preferable.

As described beforehand, the aim of the conducted research is not the validation of OSM data, but the detection of vandalism. However, the separation between these two domains is not always clear. The focus lies on vandalism detection, because OSM (and especially the editors) already incorporate different methodologies for quality assurance and validation. For example, JOSM informs a user prior to the upload if there are any intersecting geometries or duplicated elements. However, the editors only inform the user, but do not refuse to actually upload the changes.

The principle for the vandalism detection of our prototypical implementation is similar to the basic approach of a firewall: prefer the detection of too many rather than detection of too little cases of vandalism. Thus, the prototype tends to detect more vandalism cases than there actually are in reality. That way, it is assured that there are less misses, but also more false positives. However, as the system only informs about vandalism (instead of actually blocking vandalism), this is rather uncritical.

For future work, it will be important to enhance the API of the developed prototype. By providing the gathered results (*i.e.*, the detected vandalism) via a well-defined interface, other application developers can use these results for their purposes. One possible (and desirable) application is a tool

that enables users to register as a patrol for a distinct area. This way, a user can define a distinct region and/or distinct attributes and as soon as *OSMPatrol* detects a vandalism type that suites to a patrol's preference pattern, he or she is informed via. e-mail. Another potential application is a platform that enables well-known users to highlight edits as vandalism or non-vandalism and to maintain an appropriate white-list for the *OSMPatrol*.

In general, the topic of vandalism detection and prevention is also being discussed in the OSM community, e.g., limitation of the OSM API. As mentioned before, a possible solution could be to allow experienced users to review submitted content of new and, maybe, inexperienced users. This approach has already been implemented by the Google MapMaker platform [29]. However, in this case, the question remains: Are there enough volunteers available that are willing to work on some manual data validation in the future?

References

1. Diaz, L.; Granell, C.; Gould, M.; Huerta, J. Managing user-generated information in geospatial cyberinfrastructures. *Future Gener. Comput. Syst*. **2011**, *27*, 304–314.
2. Goodchild, M.F. Citizens as sensors: The world of volunteered geography. *GeoJournal* **2007**, *69*, 211–221.
3. Heipke, C. Crowdsourcing geospatial data. *ISPRS J. Photogramm.* **2010**, *65*, 550–557.
4. Neis, P.; Zielstra, D.; Zipf, A. The street network evolution of crowd-sourced maps: OpenStreetMap in Germany 2007–2011. *Future Internet* **2012**, *4*, 1–21.
5. Goetz, M. Using crowd-sourced indoor geodata for the creation of a three-dimensional indoor routing web application. *Future Internet* **2012**, *4*, 575–591.
6. Mooney, P.; Sun, H.; Corcoran, P.; Yan, L. Citizen-Generated Spatial Data and Information: Risks and Opportunities. In *Proceedings of IEEE International Conference on Intelligence and Security Informatics*, Beijing, China, 10–12 July 2011.
7. Neis, P.; Zipf, A. OpenRouteService.org is Three Times "Open": Combining OpenSource, OpenLS and OpenStreetMaps. In *Proceedings of the GIS Research UK 16th Annual conference GISRUK 2008*, Manchester, UK, 2–4 April 2008.
8. Over, M.; Schilling, A.; Neubauer, S.; Zipf, A. Generating web-based 3D city models from OpenStreetMap: The current situation in Germany. *Comput. Environ. Urban Syst.* **2010**, *34*, 496–507.
9. OpenGeoData.org. *Google IP Vandalizing OpenStreetMap*. Available online: http://opengeodata.org/google-ip-vandalizing-openstreetmap (accessed on 5 August 2012).
10. Wikipedia, the Free Encyclopedia. *Subtle Vandalism Taskforce*. Available online: http://en.wikipedia.org/wiki/Wikipedia:Subtle_Vandalism_Taskforce (accessed on 12 September 2012).
11. OpenStreetMap Wiki. *OpenStreetMap Data Working Group*. Available online: http://wiki.openstreetmap.org/wiki/Data_working_group (accessed on 5 August 2012).
12. OpenStreetMap Wiki. *OpenStreetMap Watch List*. Available online: http://wiki.openstreetmap.org/wiki/OWL_%28OpenStreetMap_Watch_List%29 (accessed on 5 August 2012).

13. ITO. *OSM Mapper.* Available online: http://www.itoworld.com/static/openstreetmap_tools/osm_mapper.html (accessed on 5 August 2012).

14. Van den Berg, H.; Coetzee, S.; Cooper, A.K. Analysing Commons to Improve the Design of Volunteered Geographic Information Repositories. In *Proceedings of AfricaGEO 2011*, Cape Town, South Africa, 31 May–2 June 2011.

15. Caminha, C.; Furtado, V. Modeling User Reports in Crowdmaps as a Complex Network. In *Proceedings of 21st International World Wide Web Conference*, Lyon, France, 16–20 April 2012.

16. Brando, C.; Bucher, B. Quality in User-Generated Spatial Content: A Matter of Specifications. In *Proceedings of the 13th AGILE International Conference on Geographic Information Science*, Guimarães, Portugal, 10–14 May 2010.

17. Chin, S.-C.; Street, W.N.; Srinivasan, P.; Eichmann, D. Detecting Wikipedia Vandalism with Active Learning and Statistical Language Models. In *Proceedings of the 4th Workshop on Information Credibility (WICOW '10)*, Raleigh, NC, USA, 27 April 2010.

18. Potthast, M. Crowdsourcing a Wikipedia Vandalism Corpus. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, Geneva, Switzerland, 19–23 July 2010.

19. Adler, B.; de Alfaro, L.; Mola-Velasco, S.; Rosso, P.; West, A. Wikipedia Vandalism Detection: Combining Natural Language, Metadata, and Reputation Features. In *Computational Linguistics and Intelligent Text Processing*; Springer: Berlin, Germany, 2011; Volume 6609, pp. 277–288.

20. Mola-Velasco, S.M. Wikipedia Vandalism Detection. In *Proceedings of the 20th International Conference Companion on World Wide Web (WWW '11)*, Hyderabad, India, 28 March–1 April 2011.

21. West, A.G.; Kannan, S.; Lee, I. Detecting Wikipedia Vandalism via Spatio-temporal Analysis of Revision Metadata? In *Proceedings of the Third European Workshop on System Security (EUROSEC '10)*, Paris, France, 13 April 2010.

22. OpenStreetMap.org. *OpenStreetMap Statistics*. Available online: http://www.openstreetmap.org/stats/data_stats.html (accessed on 8 August 2012).

23. Neis, P.; Zipf, A. Analyzing the contributor activity of a volunteered geographic information project—The case of OpenStreetMap. *ISPRS Int. J. Geo-Inf.* **2012**, *1*, 146–165.

24. Zielstra, D.; Hochmair, H.H. A comparative study of pedestrian accessibility to transit stations using free and proprietary network data. *Transp. Res. Rec.* **2011**, *2217*, 145–152.

25. Ludwig, I.; Voss, A.; Krause-Traudes, M. A comparison of the street networks of Navteq and OSM in Germany. *Adv. Geoinf. Sci. Changing World* **2011**, *1*, 65–84.

26. Girres, J.F.; Touya, G. Quality assessment of the French OpenStreetMap dataset. *Trans. GIS* **2010**, *14*, 435–459.

27. Haklay, M. How good is volunteered geographical information? A comparative study of OpenStreetMap and ordnance survey datasets. *Environ. Plan. B* **2010**, *37*, 682–703.

28. Mooney, P.; Corcoran, P.; Ciepluch, B. The potential for using volunteered geographic information in pervasive health computing applications. *J. Ambient Intell. Human. Comput.* **2012**, doi:10.1007/s12652-012-0149-4.

29. Google Map Maker. *Frequently Asked Questions*. Available online: http://www.google.com/mapmaker/mapfiles/s/faq.html (accessed on 12 September 2012).

30. OpenStreetMap Wiki. *Vandalism in OpenStreetMap Wiki*. Available online: http://wiki.openstreetmap.org/wiki/Vandalism (accessed on 8 August 2012).

31. The Bing Team. *Bing Engages Open Maps Community—Bing Maps Blog*. Available online: http://www.bing.com/community/site_blogs/b/maps/archive/2010/11/23/bing-engages-open-maps-community.aspx (accessed on 27 September 2012).

32. OSMstats. *Statistics of the Free Wiki World Map*. Available online: http://osmstats.altogetherlost.com (accessed on 11 August 2012).

33. Planet OpenStreetMap. *Index of/*. Available online: http://planet.openstreetmap.org (accessed on 1 August 2012).

34. OpenStreetMap Wiki. *Vandalismus Zwijndrecht*. Available online: http://wiki.openstreetmap.org/wiki/File:Vandalismus_Zwijndrecht.gif (accessed on 11 August 2012).

35. Potthast, M.; Stein, B.; Gerling, R. Automatic Vandalism Detection in Wikipedia. In *Proceedings of the IR Research, 30th European Conference on Advances in Information Retrieval (ECIR '08)*, Glasgow, Scotland, 30 March–3 April 2008.

36. OpenStreetMap.org. *OSM User Blocks*. Available online: http://www.openstreetmap.org/user_blocks (accessed on 1 August 2012).

37. OpenStreetMap Wiki. *Import/Catalogue*. Available online: http://wiki.openstreetmap.org/wiki/Import/Catalogue (accessed on 20 August 2012).

38. OpenStreetMap Wiki. *Automated Edits Code of Conduct*. Available online: http://wiki.openstreetmap.org/wiki/Automated_Edits/Code_of_Conduct (accessed on 20 August 2012).

39. Kittur, A.; Kraut, R.E. Harnessing the Wisdom of Crowds in Wikipedia: Quality through Coordination. In *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work*, San Diego, CA, USA, 8–12 November 2008.

40. OpenStreetMap Wiki. *OSM Map Features: Summary of Commonly Used Tags for Main Elements Used to Describe Features within OSM*. Available online: http://wiki.openstreetmap.org/wiki/Map_Features (accessed on 13 August 2012).

41. Neis, P. *How Did You Contribute to OpenStreetMap?* Available online: http://hdyc.neis-one.org (accessed on 1 August 2012).

42. OpenStreetMap Wiki. *Osmosis*. Available online: http://wiki.openstreetmap.org/wiki/Osmosis (accessed on 11 August 2012).

43. Wikipedia, the Free Encyclopedia. *Huggle*. Available online: http://en.wikipedia.org/wiki/Wikipedia:Huggle (accessed on 24 September 2012).

44. Mooney, P.; Corcoran, P. Characteristics of heavily edited objects i**n Op**en*S*treetMap. *Future Internet* **2012**, *4*, 285–305.

45. Sui, D.Z. Tobler's first law of geography: A big idea for a small world? *Ann. Assn. Amer. Geogr.* **2004**, *94*, 269–277.