International Journal of
Geo-Information

Article

A Framework for Data-Centric Analysis of Mapping Activity in the Context of Volunteered Geographic Information

Karl Rehrl * and Simon Gröchenig

Salzburg Research Forschungsgesellschaft m.b.H., Jakob-Haringer-Straße 5, 5020 Salzburg, Austria; simon.groechenig@salzburgresearch.at

* Correspondence: karl.rehrl@salzburgresearch.at; Tel.: +43-662-2288 (ext. 416)

Academic Editors: Jamal Jokar Arsanjani, Ming-Hsiang (Ming) Tsou and Wolfgang Kainz
Received: 5 November 2015; Accepted: 22 February 2016; Published: 15 March 2016

Abstract: Over the last decade, volunteered geographic information (VGI) has become established as one of the most relevant geographic data sources in terms of worldwide coverage, representation of local knowledge and open data policies. Beside the data itself, data about community activity provides valuable insights into the mapping progress which can be useful for estimating data quality, understanding the activity of VGI communities or predicting future developments. This work proposes a conceptual as well as technical framework for structuring and analyzing mapping activity building on the concepts of activity theory. Taking OpenStreetMap as an example, the work outlines the necessary steps for converting database changes into user- and feature-centered operations and higher-level actions acting as a universal scheme for arbitrary spatio-temporal analyses of mapping activities. Different examples from continent to region and city-scale analyses demonstrate the practicability of the approach. Instead of focusing on the interpretation of specific analysis results, the work contributes on a meta-level by addressing several conceptual and technical questions with respect to the overall process of analyzing VGI community activity.

Keywords: volunteered geographic information; data analysis; mapping activity; framework

1. Introduction

Over the last decade, the production of geographic information has radically changed. With the availability of cheap location technologies, Open Source GIS and the World Wide Web as a collaboration platform, an ongoing shift from professional to volunteer mappers is on the way. This phenomenon was coined as Volunteered Geographic Information (VGI) some years ago [1]. Since these early days, VGI has received continuously growing interest by individual mappers and has allowed for a wide spread of collaborative mapping activities. With the growing success of VGI, researchers have started to conduct scientific analyses with the goal to better understand the phenomenon. While early analyses primarily investigated the quality of the outcome, over the last years the activity of VGI communities itself has come into focus. While activity of online communities has been studied in different domains, e.g., in the context of Wikipedia [2,3], VGI projects such as OpenStreetMap (OSM) are a welcome playground for GI scientists due to their open access policies and well-documented community processes, e.g., mailing lists or the OSM Wiki. Moreover, the OSM project maintains a complete history of the database back to the year 2005. While most of the previous work (see Mooney *et al.* [4] for a detailed review) reveals valuable insights into activities of VGI communities, it has been found that a common conceptual and technical framework for analyzing mapping activities in the context of VGI is missing. This research gap is twofold: On the one hand, related work lacks a theoretically grounded definition of mapping activities. Some authors use the notion of "edits" but do not explicitly give a definition. Is an "edit" a new version of a feature in the change history (with the side effect that more than one information item of a feature could have been changed) or a change of a single information item of a feature (e.g., the change of an attribute value)? A clear and theoretically grounded definition of a mapping activity could support researchers and help ensure comparability of analyses results. On the other hand, each VGI project comes up with its own community-driven data model (e.g., the OSM data model) and it is not quite clear how such a data model maps onto well-established GIS models like the OGC specifications [5]. Thus, most of the activity analyses in the context of VGI tackle mapping activity on the level of project-specific data entries (e.g., OSM Nodes, Ways or Relations) rather than on a feature level following a well-established feature specification. Lifting analyses to a feature level could help to abstract from arbitrary data models by aggregating single edits to more complex feature changes. For most of the analyses, feature changes represent the necessary level of detail and thus are most likely the subject of interest. Again, a standardized mapping from project-specific data models to well-established GIS specifications could help to make results comparable and reproducible.

This work closes the gap with a conceptual and technical framework as a foundation for standardized analyzes of mapping activities. The proposed framework extends Budhathoki's work [6] with respect to the Action & Interaction Arena and builds upon the well-established concepts of *Activity Theory* [7] for modelling VGI contributions from a user's perspective. As proposed by *Activity Theory*, the work structures VGI mapping activities in *Actions* and *Operations* and proposes a mapping scheme for aggregating *Operations* to *Actions*. While *Operations* are defined as atomic concepts for manipulating database entries, *Actions* are defined as aggregated concepts for manipulating features. Thus, the framework contributes to a standardized mapping from arbitrary data models (such as the OSM data model) to the feature level (OGC Feature Specification). For evaluating the framework it is applied to historical data (full history) of the OSM project. In doing so, the work also proposes a technical frame for processing and analyzing OSM history data. For evaluation purposes, the work uses three different types of analyses, namely contribution profiling, completeness estimation and change detection. The analyses are applied to datasets of different spatial and temporal scopes (continent, country and region as well as different time periods) and answers different research questions with respect to community activity.

The remainder of the paper is structured as follows: Section 2 gives an overview of related work. Section 3 proposes the conceptual framework. Section 4 addresses technical aspects and demonstrates the applicability of the framework in the context of OSM. Section 5 evaluates the framework with different examples and Section 6 concludes the work.

2. Related Work

For several years, a continuously growing group of researchers have worked towards a better understanding of the VGI phenomenon. Budhathoki *et al.* [6] structure VGI research in three arenas, namely *motivational aspects*, questions about *action and interaction of contributors*, as well as questions about the *outcome of the activity*. While most of the researchers have analyzed the outcome of the activity, some have been concerned with the action and interaction of contributors, being the relevant studies in the context of the current work.

One of the first studies contributing to data-centric analysis of mapping activity in the context of VGI communities came from Haklay *et al.* [8]. In this study, the authors investigated the question how many volunteers it takes to map an area well. Therefore, the authors compared the quality of OSM road network data with the official dataset of Ordinance Survey and analyzed the data quality in relation to the number of contributors per square kilometer. Neis & Zipf [9] followed this research strand with an extensive analysis of the contributor activity of the worldwide OSM community. This study already took spatial as well as temporal parameters into account. In 2013, Neis *et al.* [10] compared the community development of 12 selected world regions. In addition to previous work, the authors extracted re-occurring mapper profiles. In a subsequent work, authors started to use the data on previous community development for predicting future development [11]. Contributing to the same research strand, Mooney and Corcoran [12] focused their analyses on heavily edited objects and investigated the share of contributors being responsible for a majority of edits. In 2013, the same authors published an analysis of interaction and co-editing patterns between different VGI contributors [13]. All these contributions focused on specific aspects of action or interaction, but left questions about the conceptual or technical frames for applying data-centric analysis on mapping activities open.

Some authors tackled the question from a more technical perspective. Roick *et al.* [14] proposed OSMatrix, a database for grid-based visualizations of quality indicators calculated from OSM data. Since this approach only works with static database snapshots without considering the temporal evolution, the authors extended the OSMatrix approach with a technical framework for processing and analyzing spatio-temporal quality indicators [15]. In contrast to previous work, this approach is capable of calculating quality indicators from different feature attributes including temporal aspects such as the evolution of version numbers. Although the work also tackles technical aspects, it does not explain exactly how the processing of OSM history data is being accomplished and which conceptual frame is being applied. One of the most relevant works in the context of data-centric analysis of VGI has been proposed by Barron *et al.* [16] with iOSMAnalyzer. iOSMAnalyzer is a software tool which can be used to analyze the OSM history for calculating statistics, generating diagrams and drawing maps with respect to different quality indicators. The authors give an overview of the data processing architecture, but with the lack of a theoretically grounded model. In Barron *et al.* [17], the authors complement their approach by proposing a framework with 25 quality indicators for intrinsic quality analysis of OSM data in any part of the world. The focus in this work is primarily on the quality indicators and their applicability to the OSM dataset. A first attempt towards a conceptual model for analysis of mapping activities has been proposed by Rehrl *et al.* [18]. In subsequent works [19–22], this model has been used as foundation for arbitrary spatio-temporal analysis.

The current work extends this conceptual model and proposes a theoretically grounded conceptual as well as technical framework for analyzing mapping activities in the context of VGI. The proposed framework could be applied to temporally-ordered database snapshots or complete change histories such as the OSM history, and offers a broad variety of analysis options. Different examples of continent-wide, nation-wide and regional analysis show the wide applicability as well as the efficiency of the approach. The work contributes to the research field of VGI activity analysis on a meta-level building a valuable foundation for future analyses. The following Table 1 gives a comparative overview of related work.

Table 1. Comparative overview of previous studies.

Studies	Contribution	Spatial Scope	Tool, Framework	Analysis Scope
Haklay *et al.* [8]	Relationship between number of contributors and quality of streets; 15 contributors per square mile indicate a good positional accuracy; Linus law applies to OSM	Four study areas within London	Unknown tool	Contribution profiling Completeness estimation
Neis & Zipf [9]	Contributor analysis, e.g., number of changes, when and where are contributors active	Global	Java application Open Source libraries	Contribution profiling
Neis *et al.* [10]	Contributor and data analysis in relation to area and number of inhabitants in selected cities; more contributors in European cities; many non-local contributors outside Europe	Twelve urban areas (global)	Java application Open Source libraries	Contribution profiling
Arsanjani *et al.* [11]	Raster-based analysis of OSM for predicting future developments	Heidelberg (Germany)	Cellular automata modelling	Contribution profiling
Mooney and Corcoran [12]	Analyzes the history of OSM entities with at least 15 versions; reveals disagreements between contributors on some entities	UK, Ireland	OSM History Splitter PostgreSQL	Contribution profiling Change detection
Mooney and Corcoran [13]	Contributor behavior and interaction between them; mapping behavior is not predictable; contributors can be clustered into groups; many contributors work on same entities	Seven cities (global)	Unknown tool	Contribution profiling
Roick *et al.* [14,15]	General statistics related to entities or contributors are visualized in a hexagonal grid to enable visual analytics	Six European countries	OSMatrix	Contribution profiling
Barron *et al.* [16,17]	Analyzes OSM data based on 25 intrinsic quality indicators; generates statistics, diagrams and maps	Global	iOSMAnalyzer	Contribution profiling Completeness estimation Change detection
Rehrl *et al.* [18]	First conceptual model to analyze VGI activity data	Global	Activity model	Contribution profiling

3. A Conceptual Frame for Structuring VGI Community Activity

As described in Budhathoki *et al.* [6], VGI projects are run by a few fundamental processes of community action. The key processes are (1) building a community structure; (2) describing norms and rules; (3) building and maintaining community tools and (4) mapping. Although each process is a crucial prerequisite for a successful project, the "mapping activity" itself is responsible for building the database of geographic information, which may be considered the main goal of a VGI project. Thus, by using the term "activity" in this work, we refer to "mapping activity", being aware that this is not the only activity in the context of a VGI project, but the most crucial one. Before detailing the structure of VGI mapping activities, we give a definition of the term "activity". Looking into the concepts of activity theory seems appropriate for that purpose. Having its origin in Russian psychology, in the 1990s, computer scientists adopted activity theory as a potential framework for structuring and understanding human activity in human–computer interaction [7,23–25]. Following their proposal, activity theory defines a human activity as "a form of doing directed to an object". The basic structure of an activity introducing relevant entities and their relationships as outlined by Engeström [23] is shown in Figure 1.

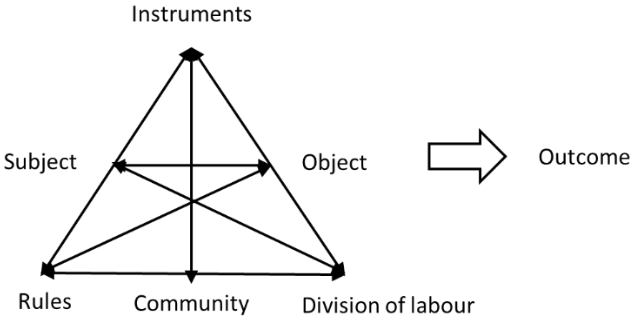


Figure 1. Basic model for structuring an activity (reproduced from [23], p. 63, with kind permission of the author).

Engeström's activity system model is well-suited for defining "activity" in the context of VGI. *Subject* denotes an arbitrary volunteer contributing to a VGI project. The *object* is the VGI project or more specifically the VGI dataset which is collected by volunteers. Kuutti [25] notes that objects could be either material objects or less tangible things or totally intangible things, which holds for VGI datasets or databases. Subjects use *instruments* (most likely devices for data acquisition as well as open source software for mapping) to manipulate the object and which are part of a *community*. The relationship between volunteers and the community is constrained by a *set of rules*. *Division of labor* characterizes the relationship between community members and the object. According to Budhathoki *et al.* [6], successful VGI projects need several sub-activities supporting the overall mapping activity of the community. These sub-activities map out the different parts of Engeström's activity system model, e.g., building a community structure, describing norms and rules or building and maintaining community tools and can be modeled as activities of its own. For example, "building and maintaining tools for data acquisition and mapping" can be modeled as a separate VGI activity, which involves individual community members (in some cases not the data contributors) and follows specific rules. In the context of this specific activity, the object is not "mapping", but building and maintaining software tools for assisting the mapping activity. We assume that activities 1 to 3 are necessary pre-requisites of any VGI project, building and maintaining a meaningful context for the mapping activity itself. OpenStreetMap (OSM), for example, clearly reveals the different activities of a VGI project. The main activity is "mapping" with the object to "map the world". The outcome is the OpenStreetMap dataset, also called "the map". Subjects are the members of the OSM community. Building community structures is a separate activity with dedicated actions (e.g., organizing mapping parties), running in parallel and supporting the mapping activity. Norms and rules are described in the OSM Wiki thus "building and maintaining the OSM Wiki" is another activity in support of the definition of mapping rules. "Building and maintaining community tools" is also defined as separate activity, running in parallel to mapping and heavily supporting the mapping activity. Without community tools it would be impossible to run mapping activities by volunteers. In OSM, for example, there exist numerous tools for collecting data, mapping features or submitting features to the database.

While this section introduced activity theory as anchor for structuring VGI community activity and helped to break down the overall activity of a VGI project in several sub-activities, the next section focuses on structuring the sub-activity "mapping".

3.1. Structuring VGI Mapping Activity

For structuring the activity "mapping" it is necessary to have a closer look at the different levels of activity theory. Since activities are often complex formations, activity theory helps to structure the transformation process towards the outcome on different hierarchical levels. These levels are *activity*, *action* and *operation*. Figure 2 shows the hierarchical relationship.

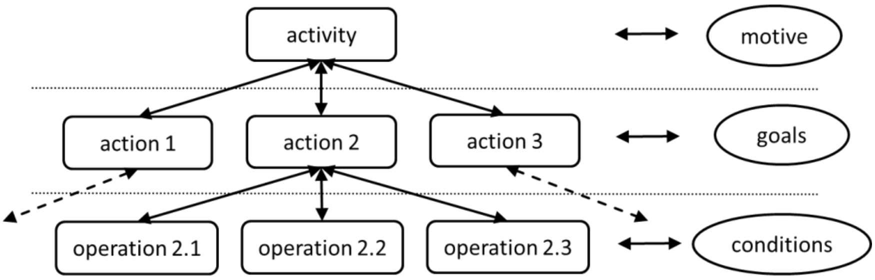


Figure 2. Hierarchical structure of an activity (reproduced from [26], Figure 3.4, © Massachusetts Institute of Technology, by permission of The MIT Press).

Activities are driven by motives. The most general motive in VGI is "mapping the world". However, it is most likely that numerous sub-motives like "mapping a country", "mapping a city", "improving positional accuracy", "improving thematic accuracy" and many more exist. Each activity can be broken down into a chain of subsequent actions. Whereas activities are driven by motives, actions are driven by concrete goals. Typical examples in the context of mapping activities are "mapping of a geographic features" like "mapping a building", "mapping a street" or "mapping landuse". Actions could be further broken down into sequences of operations. Operations denote the lowest level of human activity and thus could not be further split up. Thus, operations may be considered atomic. Typical examples of operations in the context of VGI are "setting the coordinates of a point feature", "adding an attribute to a feature" or "changing an attribute value of a feature". Table 2 outlines the different levels of activity theory and shows their applicability with respect to VGI mapping activity.

Table 2. Concepts of Activity Theory applied to VGI mapping activities [18].

	Activity Theory	Applied to VGI Mapping Activity
Activity	A chain of associated actions following the same motive.	A set of mapping actions driven by a certain motive (e.g., improving accuracy).
Action	A goal-driven part of an activity. An ordered sequence of operations.	A sequence of operations by a single volunteer within a certain timespan manipulating a single feature to reach a certain goal (e.g., editing a feature geometry).
Operation	Atomic. Being part of an action. Being executed automatically in order to reach a certain goal.	Atomic change to a single feature by a unique volunteer and at a unique timestamp (e.g., updating coordinates).

3.2. Operations

The three levels of activity theory (activity, action, operation) can be used as a powerful and universal conceptual frame for structuring mapping activity. Starting at the lowest level, the so called *operation level*, a distinct set of operations for data manipulation within a VGI project can be defined. For accurately specifying different VGI operations, they can be attributed with distinct properties such as *operation type*, *timestamp* (the time at which the operation has been executed), a unique *user identification* (the contributor executing the operation) and a unique *feature identification* (the feature being manipulated by the operation). Such an operation set, especially the operation types, clearly depend on the specific concepts and data models used within a VGI project and thus have to be defined separately for each project. An appropriate way for identifying data manipulation operations in the context of a VGI project is to perform a so called *task analysis* as proposed by Timpf [27]. In the context of a VGI project, task analysis reveals typical mapping tasks, which are taken over by community members. In order to break down these tasks to the operation level, it is necessary to either have a look at the GIS-related functions of data acquisition tools or at least at the available database operations, which are used for manipulating the data. A common paradigm for structuring data-related operations is the CRUD paradigm [28], which is well-known from the world of databases. Databases typically offer four atomic operations for accessing and manipulating data, namely *Create*, *Read*, *Update* and *Delete* (CRUD). As shown in previous work [18], the concepts of the CRUD paradigm applied to a geospatial data model used in a VGI project result in a well-defined atomic operation set for accessing and manipulating geospatial data items which could be defined for any VGI project. It has to be noticed that READ is not a data manipulation concept and thus is not considered further.

3.3. Actions

While the operation set defines atomic operations for manipulating data entries of a geospatial database and therefore heavily depends on the specific data model of a VGI project, the next activity level, the so called *action level* already provides an aggregated view on mapping activities. Actions are defined as operation sequences expressing possible ways to manipulate characteristics of *geographic features*, independently of underlying database models. As outlined in previous work [18], the definition of the action set is also based on the CRUD paradigm. For anchoring actions on the level of geographic features, it is useful to rely on a common feature model such as the *OpenGIS Abstract Specification, Topic 5: Features* [5] specified by the Open Geospatial Consortium (OGC). The OGC specification introduces the concept of a *geographic feature* as representation of a geographic phenomenon. Each feature is composed of a *geometry*, which is bound to a *spatial reference system* and a *feature type*. A *feature type* defines a set of attributes for describing the feature. Relying on this specification, the two most relevant concepts for feature geometries are *Point* and *LineString*. The specification proposes two sub-concepts for structuring *LineString,* namely *Line* and *LinearRing*. Moreover, the specification defines *Polygon* as distinct geometrical concept. Since the *Features* specification defines features as flat structures (other features may not be part of a feature), a second specification, namely the *OpenGIS Abstract Specification, Topic 8: Relationships between Features* [29] is necessary for expressing *relationships*. Combining the three geometrical concepts *Point*, *LineString* (used as aggregate for *Line* and *LinearRing*) and *Polygon* and the *Relation* concept with the data manipulation concepts of the *CRUD* paradigm (*Create, Update, Delete*) results in 12 distinct actions for manipulating geographic features (note that *Read* is not considered since it is not a concept for manipulating features) (Table 3).

While operations are specific to the data structure of the dataset, the same actions can be used regardless of the data structure. Since most VGI mapping activity analyses are conducted on the action level, a well-defined ruleset for mapping operation sets to the action set has to be defined. The ruleset for mapping OSM-specific operations to the generic level of VGI actions is outlined in Section 4.4.

Table 3. Create, update and delete action types for each geometry type.

	Create	Update	Delete
Point	Ac Create Point	Ac Update Point	Ac Delete Point
LineString	Ac Create LineString	Ac Update LineString	Ac Delete LineString
Polygon	Ac Create Polygon	Ac Update Polygon	Ac Delete Polygon
Relationship	Ac Create Relationship	Ac Update Relationship	Ac Delete Relationship

3.4. Activities

While VGI actions express sequences of operations applied to individual features, VGI activities are defined as sequences of actions manipulating selected features with the intention of contributors following a distinct motive. For example, all actions contributing to the mapping of buildings in a city could be aggregated to one mapping activity. Alternatively, all the actions of one contributor could be grouped to one mapping activity. Thus, the main goal of modelling mapping activities is to filter or group action sets for further analyses. Research questions, for instance, are typically formulated on the activity level. Nevertheless, the actual analyses are always conducted on the action level.

4. Analyzing Mapping Activity in the Context of OpenStreetMap

Before structuring mapping activities of the OSM community, it is worth having a closer look at the data model of OSM. The data model consists of three different concepts called *Nodes*, *Ways* and *Relations*. *Nodes* are the only entities which are anchored to a geographic reference frame (Euclidean space). *Ways* are lines or simple polygon entities, composed from an ordered sequence of *Nodes*, either open or closed. *Relations* can be used to define typed relationships between *Nodes*, *Ways* and other *Relations* and are used to represent more complex geometric elements like multi-polygons. For giving some meaning to these entities, the concept of *Tag*s is used. *Tags* are textual key-value-pairs, which can be attached to *Nodes*, *Ways* and *Relations*. A more detailed description of the OSM data model can be found in Ramm and Topf [30].

Mapping activity in OSM follows the overall motive to "map the world". The outcome of the "mapping activity" is the so called "map", which is a database under the Open Database License (ODbL). Due to the open nature of the project, any registered OSM contributor is allowed to modify data stored in the database without any restrictions. For submitting or manipulating data, the project defines a basic set of database operations, following the CRUD paradigm [28]. This means that any of the entities defined in the data model (*Nodes*, *Ways* and *Relations*) may be *Created*, *Read*, *Updated* or *Deleted*. Specific to OSM is the used data model which makes it necessary to define OSM-specific operations.

4.1. Operations in OSM

First, the possible operations for manipulating entries in the OSM database have to be defined. An *OSM Node* may be created, *Tags* may be added, modified or removed, *Coordinates* may be updated and each *OSM Node* may be deleted again. An *OSM Way* may also be created, *Nodes* may be added, reordered or removed (since a *Way* is composed of an ordered sequence of *Nodes*), *Tags* may be added, modified or removed and each *Way* may be deleted again. An *OSM Relation* may be created, *Members* may be added, reordered or removed (*Members* can be *Nodes*, *Ways* or other *Relations*), the *Role of Members* may be modified, *Tags* may be added, modified or removed and each *Relation* may be deleted again. In addition to these operations, the OSM database also allows recreating deleted versions of *Nodes*, *Ways* or *Relations*. The complete set of basic OSM operations is shown in Table 4. These basic set of operations may be used for expressing any change to the database back to the early days of the project. It is worth mentioning that there are also operations for reading the data, but since these operations do not change database entries, they are not considered for this work.

Table 4. Overview of different operation types being executable on the different OSM data types (Node, Way, Relation).

	Node	**Way**	**Relation**
Create	Op Create Node	Op Create Way	Op Create Relation
Update (geometry)	Op Update Coordinate	Op Add Node Op Remove Node Op Reorder Node Op Update Coordinate	Op Add Member Op Remove Member Op Reorder Member Op Update Coordinate
Update (attribute)	Op Add Tag Op Update Tag Value Op Remove Tag	Op Add Tag Op Update Tag Value Op Remove Tag	Op Add Tag Op Update Tag Value Op Remove Tag
Update (entity)	Op Recreate Node	Op Recreate Way	Op Recreate Relation
Delete	Op Delete Node	Op Delete Way	Op Delete Relation

Each operation is attributed with a *reference to the OSM entity* (unique identifier), the *operation type*, the *contributor ID* executing the operation, the *timestamp* and the *changeset identifier*. Optional operation attributes include the *coordinate* (for node operations), a *key-value pair* (for tag operations), a *reference-identifier* (for add/remove/reorder operations) and a *position-pointer* (also for add/remove/reorder operations).

In the context of OSM, a line split results in a new *Way* and the respective *Nodes* are removed from the existing *Way* and added to the new one. This situation can be recognized from the data due to the unique identifier of the referenced *Nodes* and *Changeset*s. Contrarily, a line merge is detected if the *Nodes* which are removed from a deleted *Way* are added to another *Way* within the same *Changeset*.

To show the applicability of the operation set, the following example (Figure 3) shows a set of atomic OSM operations representing a simple mapping action. The goal of the OSM mapper was to map a new residential street with four *Nodes* (red line in Figure 3). Three of those *Nodes* had to be created while one *Node* already existed as part of another street. The creation of the three *Nodes* results in three *Op Create Node* operations. During the creation each *Node* gets a unique identifier and a *Coordinate*. The creation of the street entity (*Way*) is represented by an *Op Create Way* operation. During creation the *Way* gets a unique identifier. Adding a geometry to the *Way* implies attaching the four *Nodes* (the three new ones and the one existing) in the correct sequence to the *Way*. This procedure is represented by four *Op Add Node* operations. To define the *Way* as a residential street, an *Op Add Tag* operation adds an attribute with "highway" as *key* and "residential" as *value*. If the contributor adds further information to the *Way* (e.g., a name or a speed limit) additional *Op Add Tag* operations will be added. Table 5 summarizes the atomic OSM operations representing the creation of a new street.

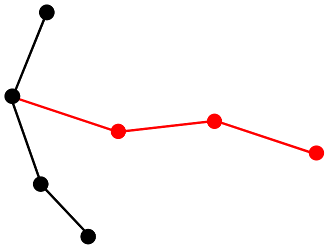


Figure 3. A new residential street (red) being created and connected to an existing one (black).

Table 5. Operation sequence of a single mapping action (creating a street).

Entity ID	Operation Type	Contributor ID	Timestamp
Node 5	Op Create Node	VGI Mapper	2015-07-09 09:05:36
Node 6	Op Create Node	VGI Mapper	2015-07-09 09:05:36
Node 7	Op Create Node	VGI Mapper	2015-07-09 09:05:36
Way 2	Op Create Way	VGI Mapper	2015-07-09 09:05:36
Way 2	Op Add Node (Node 2)	VGI Mapper	2015-07-09 09:05:36
Way 2	Op Add Node (Node 5)	VGI Mapper	2015-07-09 09:05:36
Way 2	Op Add Node (Node 6)	VGI Mapper	2015-07-09 09:05:36
Way 2	Op Add Node (Node 7)	VGI Mapper	2015-07-09 09:05:36
Way 2	Op Add Tag (highway = residential)	VGI Mapper	2015-07-09 09:05:36

4.2. Extracting Operations from the OSM Full History

A common source to generate operation sets for mapping activities related to OSM is the OSM Full History File (http://planet.openstreetmap.org/pbf/full-history/). The OSM History lists all current and previous OSM entities (*Nodes*, *Ways* and *Relations)* with all versions back to the launch of the OSM project. For analyzing up-to-minute data, it is possible to add the latest data from the minutely, hourly or daily *Changesets*. Entities are ordered by *type* (*Nodes* first, than *Ways* and *Relations*), *unique identifier* and *version* in ascending order. In other words, the history file represents all historical versions of OSM database entries as time series, but does not represent changes between two consecutive versions of the same entity. Thus, operations have to be generated by comparing two consecutive versions of one entity. This task has to be repeated for all versions of all entities. Figure 4 outlines the overall workflow for extracting operations from the OSM Full History File.

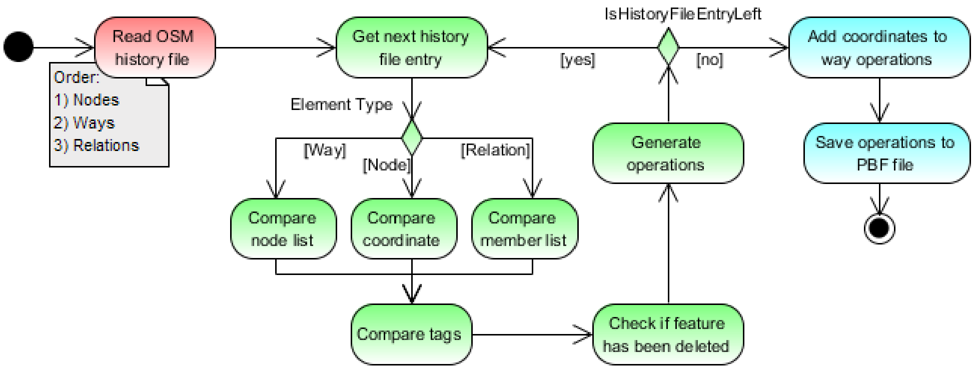


Figure 4. Workflow for extracting operations from the OSM Full History File [18].

During the extraction process, the operation generator receives each version of an OSM entity consecutively along with the previous version of the same entity (except the first version of an entity). The operation generator compares two consecutive versions and identifies changes. These changes are mapped to the corresponding operations from the operation set (see Table 4). A detailed description of the whole process is given in previous work [31]. Applying the operation generator to the OSM Full History File published in September 2015 consisting of 3,812,543,515 entities (*Nodes, Ways and Relations*) results in 14,157,025,062 atomic operations. They are stored in optimized Protocol Buffer Format (PBF) files and organized in batches due to the high number. The size of one PBF file containing one batch is set to 5 megabytes (variable). The operations within the PBF files are ordered by *entity type* (following the order *Node*, *Way*, *Relation*), *unique entity identifier* and *timestamp*. If successive operations share common attributes (e.g., entity identifier, operation type), only the first entry stores this information in order to save storage space. Since operations may not be subject to change due to their atomic nature, they are perfectly suited for acting as intermediate data format for further analysis. Instead of storing operations in PBF files, it is also possible to store operations in databases.

It has to be noticed that specific characteristics of the OSM data model play a major role during the generation of operations. Since only *OSM Nodes* contain geographic references, the geometry of any linear or polygonal feature (modelled as *Way* or *Relation*) is represented by an *OSM Node* sequence. If the geometry of a linear or polygonal feature is changed, also the associated *OSM Nodes* get new versions in the history file in case of coordinate changes. The operation generator is looking for these coordinate changes and adds *Op Update Coordinate* operations to the feature.

4.3. Spatial Indexing

Up to this point, the operations are ordered by (OSM) entity type and entity ID (ascending). Hence, the data is spatially disordered. While this ordering is suitable for global analysis, it is very inappropriate for analyzing sub-regions like countries or cities. In order to find features within a region, an efficient spatial index structure has been adopted.

A point region quadtree is used to divide the world into quadrants [32]. The overall bounding box is defined by the minimum and maximum values of the coordinate reference system WGS84, which is also used by OSM. This quadtree recursively subdivides areas with many features into sub quadrants. Consequently, data in urban areas is represented by several sub quadrants whereas data in rural or uninhabited regions is represented by only few quadrants. It has been empirically founded that a quadrant capacity of 100,000 features fits best the requirement of efficient processing time. If a quadrant exceeds its capacity, it is going to be subdivided. The maximum quadtree level has been defined with 10, meaning that a quadrant cannot be sub-divided more than 10 times. While all other quadrants are constrained by a feature capacity, quadrants on the last level offer unconstrained capacity. The maximum tree level has been introduced because a smaller quadrant size at the deepest level has been found inefficient during analysis. A level-10-quadrant has a height of 20 km and a width of 10–40 km (depending on latitude), which proved to be sufficient for efficiently extracting operations for cities or regions. Within a quadrant, the operations are stored in separate PBF files and ordered by (OSM) entity type and entity ID. During the building process of the spatial index, each feature is inserted to the smallest quadrant enclosing its bounding box without overlapping the quadrant. The bounding box of a feature covers all coordinates belonging to the associated operations.

Figure 5 shows the quadtree structure for the operations generated from the September 2015 OSM Full History File. The colors indicate the number of features within a quadrant. Light colors mean that only a few features are stored within the quadrant while dark colors indicate a high number of features (up-to 100,000). If more operations are added to the quadtree, dark-colored quadrants would be the first candidates to be subdivided into sub quadrants. Grey-colored quadrants (as in Europe, Japan or Eastern USA) contain already more than 100,000 features and therefore may not be sub-divided again. The quadtree structure has been preferred over a grid structure due to the unequal distribution of operations (e.g., in sparsely populated areas like oceans or deserts). The access of many files with a low number of features would result in a poor reading performance.

For selecting the features of a sub-region like a country, city, or continent for further analyses, the topological relationship (intersect) between the quadrants and a filter polygon has to be considered. Quadrants being *inside* the filter polygon are processed without exceptions. Quadrants being *disjoint* from the filter polygon are skipped. Quadrants *overlapping* the filter polygon are processed in more detail as the geometries of all included features have to be intersected with the filter polygon in order to identify features being located inside the selected region.

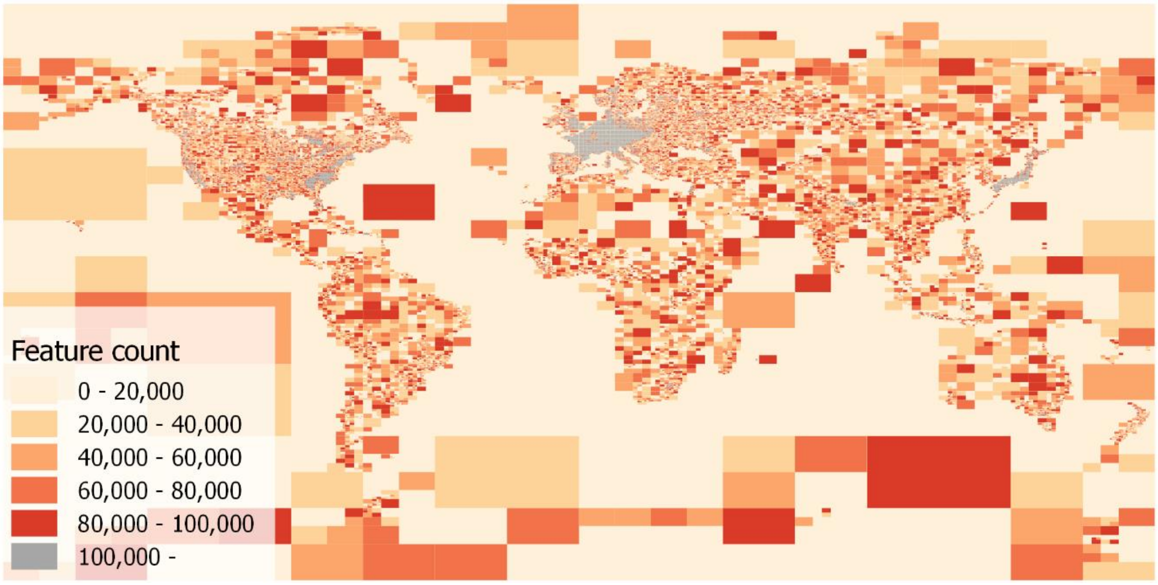


Figure 5. Quadtree structure representing the spatial indexing of the operations generated from September 2015 OSM Full History File (colors indicate the number of features within a cell).

4.4. Aggregating Operations to Actions

While operations are useful as an intermediate format due to their atomic nature, for many analyses operations are too detailed. Thus, it is useful to aggregate operations to higher-order actions being a more suitable structure for fast analyses. One such analysis task is, for example, how features evolved over time and which actions where committed by which contributor. Another argument for mapping operations to actions is in the feature-based format. While operations are dependent on the data model of a specific VGI project (e.g., OSM data model), actions are anchored to geographic features, independently of the originating data model. This ensures analyses are executed on a feature-typed model instead of the OSM model which is a crucial argument for standardized and interpretable analyses results.

Before operations may be mapped to feature-related actions, they have to be grouped to operation sequences. The following constraints for operation sequences have been defined:

(1) All operations within one sequence refer to the same OSM entity
(2) The operations within one sequence have been contributed within a time span of 24 h
(3) The operations within one sequence have been contributed by the same OSM contributor.

In other words, an operation sequence only contains operations for a unique entity (identified by a unique identifier such as the *OSM Node ID* or the *OSM Way ID*), being executed by a single contributor within 24 h. It should be mentioned that the timespan may be subject to change although experiments with different timespans (6 h, 12 h) did not reveal significant differences in the resulting action set. The generation of actions is performed per operation sequence. As introduced in Section 3.3, action types are again classified using the CRUD paradigm (*Create*, *Update* and *Delete*). Create and Delete Actions always depend on one specific feature type.

Ac Create Point/LineString/Polygon/Relationship: If a feature is created or if the tags are modified by adding a feature type to a feature, an *Ac Create* action is generated. For generating this action an *Op Add Tag* operation representing a feature type or an *Op Update Tag Value* operation which changes a regular tag to a feature type tag has to be contained in the operation sequence. Beside the *Op Add Tag* operation with the primary key and thus specifying the feature type, a create action may include several additional operations (e.g., further *Op Add Tag* operations or *Op Add Node* operations if the feature is a *LineString* feature or *Op Add Member* operations if the feature is a *Polygon* feature or a *Feature Relationship*). If these operations belong to the same operation sequence, they are aggregated to one single action.

The following example shows how a modified tag can trigger an *Ac Create* action: The OSM list of feature types defines "minor street" and includes the tag "highway" with the possible values "tertiary", "unclassified", "residential" and "service", while the value "track" is not included. If a contributor changes a features highway tag from "track" to "residential", an *Ac Create* action is generated (e.g., *OSM Way* 27154461). If the "track" value would be covered by any other feature type, an *Ac Delete* action would be generated consequently.

Ac Delete Point/LineString/Polygon/Relationship: If a feature is deleted from the database or if the tags are modified in a way that the feature type is removed, an *Ac Delete* action for a feature is generated. This requires an *Op Remove Tag* operation representing a feature type or an *Op Update Tag* Value operation which changes a feature type tag to a regular tag. The *Ac Delete* action may be joined by other operations which include the removal of additional tags (*Op Remove Tag*), nodes for way features (*Op Remove Node*) and members for relationship features (*Op Remove Member*).

Ac Update Point/LineString/Polygon/Relationship: For generating an update action it is necessary that the feature has been created before (*Ac Create* action) and is still visible (it has not been deleted). *Ac Update* actions are generated from operation sequences containing the following operation types related to the geometry or the attributes of a feature: *Op Add Tag*, *Op Remove Tag*, and *Op Modify Tag Value*. Point or LineString actions may also contain *Op Update Coordinate* operations. LineString actions may also contain *Op Add Node, Op Remove Node* operations. Polygon or Relation actions may also contain *Op Add Member, Op Remove Member* or *Op Update Role* operations.

Table 6 outlines a set of rules for mapping OSM operations to actions. The geometry type of the feature type defines which set of action mappings is used. The operation types in the mapping table indicate which operations are added to the action. If not all operations of the operation sequence are added to the action or if the feature has been updated, another *Ac Update* action is generated. If a feature is represented by multiple feature types, actions are generated for each feature type, respectively.

Table 6. Mapping of operations to Actions.

Point	LineString	Polygon	Relationship
Ac Create Point	Ac Create LineString	Ac Create Polygon	Ac Create Relation
Op Create Node Op Add Tag	Op Create Way Op Add Tag Op Add Node	Op Create Way Op Create Relation Op Add Tag Op Add Node Op Add Member	Op Create Relation Op Add Tag Op Add Member
Ac Delete Point	Ac Delete LineString	Ac Delete Polygon	Ac Delete Relation
Op Delete Node Op Remove Tag	Op Delete Way Op Remove Node Op Remove Tag	Op Delete Way Op Delete Relation Op Remove Node Op Remove Member Op Remove Tag	Op Delete Relation Op Remove Member Op Remove Tag
Ac Update Point	Ac Update LineString	Ac Update Polygon	Ac Update Relation
Op Add Tag Op Update TagValue Op Remove Tag Op Update Coordinate	Op Add Tag Op Update Tag Value Op Remove Tag Op Add Node Op Remove Node Op Reorder Node Op Update Coordinate	Op Add Tag Op Update Tag Value Op Remove Tag Op Add Node Op Remove Node Op Reorder Node Op Add Member Op Remove Member Op Reorder Member Op Update Role Op Update Coordinate	Op Add Tag Op Update Tag Value Op Remove Tag Op Add Member Op Remove Member Op Reorder Member Op Update Role Op Update Coordinate
Example feature type: address	Example feature type: street	Example feature type: building	Example feature type: route

To explain the aggregation from operations to actions, the above example outlined in Section 4.1 is used again. Table 5 outlines the operations being generated during the creation of the residential street with four nodes. The street feature should be attributed with the feature type *street* which implies a mandatory tag with the key "highway" in OSM. From an action/feature perspective, there is only one feature created, namely a residential street feature. Following the mapping rules above, all operations outlined in Table 5 are aggregated to one *Ac Create LineString* action. Since the created nodes are only used as geometrical anchors of the street and are not tagged as features, they are only considered as part of the create action and not as separate actions.

Generating actions with the aforementioned mapping rules from the world-wide operation set for the feature type "street" results in 61,194,708 *Ac Create* actions, 109,919,435 *Ac Update* actions and 8,795,477 *Ac Delete* actions (OSM History File from 14 September 2015). In total, the 179,909,620 actions have been contributed by 311,413 community members and are assembled from 1,697,262,904 operations meaning that on average 9.4 operations are aggregated to one action for the feature type "street". This data reduction has significant advantages for data analysis performance.

4.5. Analyzing Mapping Activity

After generating and indexing operations, they are prepared for further analyses. Any analysis typically starts with reading the operations from the PBF files. The plain PBF files are used for global analyses while the spatially indexed files are used for spatially constrained analyses. Applicable filters include *temporal filters* (e.g., all operations until 1 July 2015), *spatial filters* (e.g., Europe), *tag filters* (e.g., only features which possess a tag with the key *highway*) or *feature type filter*s (e.g., streets). A feature type filter typically is defined by a list of primary tags (which tags have to be attached to an OSM entity in order to declare it as a feature of the respective feature type), a list of attributive tags (which tags store attributes of the feature) and a geometry type (point, line or polygon). For example, if the tag filter is initialized with the feature type "buildings" and the tag filter is set to "house numbers" than all building features are read while only buildings with house numbers are analyzed.

The tool is also looking for line merges and splits, which often arise from modifying network data (e.g., streets). For every *Node* which has been moved to another entity during a split or merge event, an *Op Split Way* or *Op Merge Way* operation is generated. Optionally, this step can be switched off as it requires additional processing power, especially while analyzing large regions.

After aggregating operations to actions (see previous section), the resulting feature actions are analyzed. Analysis settings include a list of analysis methods and the temporal resolution (year, month or day) of the results. Analyses on an annual basis are faster, but more inaccurate, while analyses on a monthly or daily basis are more accurate, but take more time and memory. The analyzer investigates actions, contributors, feature types and other aspects and writes the resulting values into Comma Separated Values (CSV) files or a database. These files contain raw analysis results and are used for further processing or interpretation.

4.6. Performance

The generation of VGI operations and creation of the spatial index takes 140 h on a workstation computer with 3.6 GHz, 16 GB RAM and SSD storage if the whole dataset is processed or less time if a spatial filter is applied. However, generating operations and building the spatial index has to be done only once for an input dataset (e.g., OSM Full History File) in case it has been updated. Contrarily, analyzing the data including the generation of actions is done for each analysis separately as settings may change. Processing time for this second part depends on the settings and especially on the spatial filter. The spatial index structure ensures that features being located in quadrants outside of the filter polygon are not read which ensures faster results. The processing times for different kinds of analyses are mentioned in the next section.

5. Evaluation of the Framework

The following section gives examples where the framework has been used to study mapping activity. To show the wide applicability, studies with varying research questions as well as research background have been selected. The examples cover different geographic and temporal scales. On the one hand, the analyses are performed on continent (e.g., completeness estimation), country, region (e.g., contribution analysis) and city scale (e.g., change detection) and, on the other hand, either for a single point in time (e.g., completeness estimation) or a time period (e.g., change detection). All evaluation examples use the same operation set stored in the PBF files described in Section 4.2.

5.1. Contribution Profiling

One of the re-occurring questions in VGI projects is the question concerning contributions of community members. While some of the questions may be answered from official OSM statistics (http://wiki.openstreetmap.org/wiki/Stats), more fine-grained analyses (e.g., selected regions, time periods or single feature types) are not. The proposed framework provides the possibility for fast analyses of arbitrary regions and time periods. While the official contribution statistics are usually based on the OSM data model (*Node/Way/Relation*), the proposed approach allows for analyses on the feature type level following the specifications from the Open Geospatial Consortium [5]. For the following evaluation examples, mapping activities from the city of Berlin as well as the country of Nepal including the city of Kathmandu have been investigated. Nepal and Kathmandu have been selected due to a huge mapping effort in 2015 which has been started by the HOT team after the earthquake disaster that hit this region on 25 April 2015.

The first analysis answers the question which types of features have been created/updated/deleted by VGI contributors in a certain region (city of Berlin) during a certain time period (three six-month-periods). The used feature type definitions have been derived from the map features list outlined in the OSM Wiki (http://wiki.openstreetmap.org/wiki/Map_Features). Instead of simply counting database changes, the proposed model shifts the analysis on the action and feature level. Table 7 shows the number of actions in Berlin per feature type for three six-month periods between January 2014 and June 2015. For this example only OSM *Node* and OSM *Way* entities are considered while relations are ignored leading to the fact that e.g., multi-polygons of natural features may not be included. The analysis takes six minutes. The figures give a first indication of the mapping activity in the selected region. While the mapping of natural point-features has increased during the analyzed time periods, the mapping of streets or amenities has been rather constant and the mapping of buildings has increased and decreased again.

Table 7. Number of actions per feature type and six-month periods in Berlin.

Feature Type (Geometry Type)	January–June 2014	July–December 2014	January–June 2015
Building	156,831	189,119	89,581
Street	38,417	34,232	35,453
Street-Point (e.g., crossings)	2133	3029	5192
Natural-Point (e.g., tree)	4460	23,832	41,359
Natural-Polygon (e.g., lake)	1703	931	1567
Amenity-Point (e.g., hotel as point)	6249	10,821	8426
Amenity-Polygon (e.g., hotel as polygon)	3633	4189	3778
Railway	4152	4951	6911
Barrier	2772	1387	6515
Landuse	7970	6406	5023
Others	9273	13,940	10,555
Total	237,593	292,837	214,360

A second analysis answers the question about the distribution of action types (Table 8). While for the first analysis only the total number of actions per feature type has been considered, for this analysis a more fine-grained analysis of action types has been applied. *Ac Update* actions are further split into attribute-related updates (*Ac Update Attribute*) and geometry-related (*Ac Update Geometry*) updates [33]. This analysis is accomplished by defining two different update action types where each of them only contains the attribute-related respective to the geometry-related operation types. Geometry updates include not only node list changes (which can be derived from the OSM data directly) but also modified node coordinates as well as split and merged features. The modified action definitions are also responsible for the different total values in Tables 7 and 8. The numbers listed in Table 8 represent the number of actions per action type for the three six-month periods. This analysis impressively shows the strengths of the proposed approach: the flexible mapping of operation to action types allows for individual analyses with only changing a few mapping rules.

Table 8. Number of actions per action type during six-month periods for the city of Berlin.

Action Type	January–June 2014	July–December 2014	January–June 2015
Ac Create	89,219	120,507	70,184
Ac Update Attribute	76,512	89,663	77,300
Ac Update Geometry	78,750	79,076	67,051
Ac Delete	9961	14,839	8293
Total	254,442	304,085	222,828

Contributor analyses are also a suitable measure to analyze mapping activities being coordinated by specialized VGI communities like the Humanitarian OSM Team (HOT) (https://hotosm.org). HOT has developed as a major community effort in the context of crisis mapping. Typically, HOT starts its mapping efforts immediately after a (environmental) disaster. Volunteers help in mapping streets and other features in the affected region following the goal to provide actual and complete map data for rescue teams. On 25 April 2015, the country of Nepal was hit by a major earthquake. Immediately after the earthquake, HOT started an initiative for improving OSM data in Nepal. The success of this effort is shown clearly in Figure 6 detailing the number of daily contributors as well as daily actions per contributor in Nepal before and after the earthquake. The contributor statistics has been derived by using the proposed framework with a spatial filter around the Nepalese border including all feature types. The figures not only show a significant increase of contributors immediately after the earthquake, but also a significant increase of actions per contributor. Interestingly, although the number of contributors continuously decreased after the earthquake, the number of actions per contributor remained on a higher level than before the earthquake indicating that still a few people are responsible for a majority of the mapping activity. In total, 8176 contributors contributed 2,683,393 actions to the crisis mapping effort after the earthquake. Of those, 5598 (68%) had not edited the map before. Calculating the number of contributors and actions for all feature types for the whole country of Nepal took eight minutes processing time.

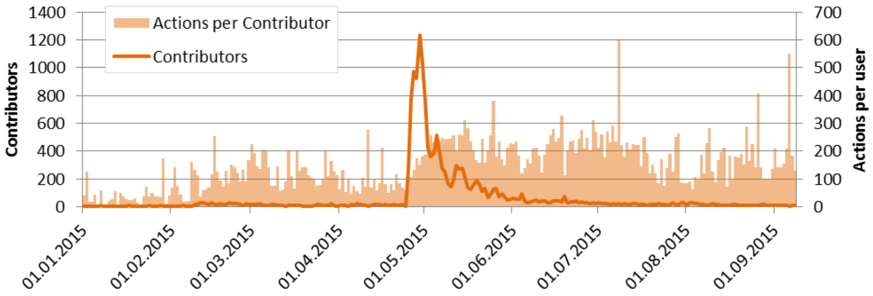


Figure 6. Number of daily contributors and actions per contributor in Nepal before and after the earthquake in April 2015 for all 26 feature types as described above.

A detailed tag analysis on the operation level allows for more detailed insights on contributor activity (Table 9). For this analysis, all attribute-related operations (*Op Add Tag*, *Op Update Tag Value* and *Op Remove Tag*) associated to actions with the feature type "building" have been analyzed. Temporal and spatial filters have been set to include all actions after the earthquake in Nepal. The data reveals that 1,701,551 new "building" tags have been created and that most created buildings are not tagged with additional attributes. Many former "area" tags have been removed in what can be interpreted as quality improvement due to the fact that, in OSM, buildings are interpreted as a polygonal feature by default. The data also reveals that 1,597,974 (94%) of all added buildings are only tagged with the value "yes" while some others are tagged as "house" (82,233) or "residential" (17,265).

Table 9. Tag operations for feature type building after the earthquake in Nepal.

Tag Key	Op Add Tag	Op Update Tag Value	Op Remove Tag	Total
building	1,701,551	10,138	59,541	1,771,230
source	98,318	102	2242	100,662
landuse	14,610	235	2528	17,373
building:levels	13,666	3	340	14,009
area	2738	0	9167	11,905
building:adjacency	8764	0	207	8971
building:use	8729	177	59	8965
shape:plan	8753	1	205	8959
roof:material	8705	2	154	8861
others	145,201	168	11,408	156,777

It might be expected that the mapping of building conditions is one of the primary goals of community activity in the context of post-earthquake mapping. A detailed contribution analysis revealed the following building condition related attributes: damage (123 buildings), damage:event (618), physical:condition (8668), collapsed (75) and condition (3). In relation to 1.7 million buildings, which have been mapped in Nepal during the aftermath of the earthquake, for only a very low number of building features the physical conditions have been acquired. The following map shows the sparse mapping of building condition related attributes.

The map reveals that the affected building features are widely distributed. While features with the attribute *damaged:event* occur in all affected regions, features with the attributes *collapsed* or *damaged* are rather concentrated. The attribute *physical:condition* has only been collected by three contributors in the city of Bharatpur which is highlighted in Figure 7. The detailed map also visualizes the condition level (good, average, and poor). The analysis clearly demonstrates that the goal of post-earthquake mapping with respect to building conditions has not been met, although companies offered post-earthquake satellite imagery showing destroyed buildings (https://www.mapbox.com/blog/nepal-earthquake-imagery-0427/). One reason could be that so far there is no accepted proposal on how to model building conditions in OSM.

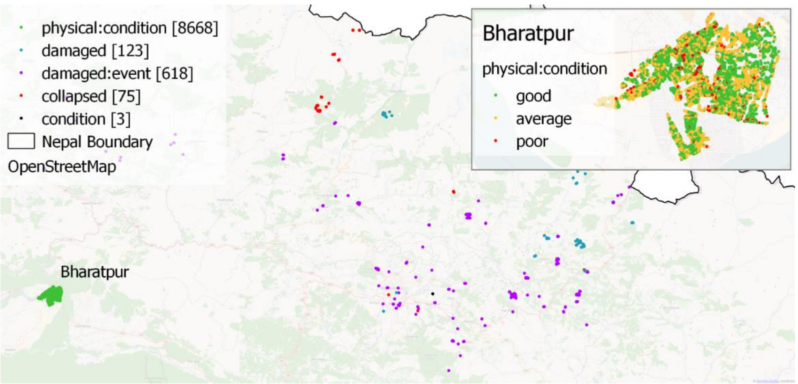


Figure 7. Map showing the location of building features in Nepal possessing a condition attribute.

5.2. Estimating Completeness

A predominant question in the context of VGI is concerned with quality assessment. Although in the past the majority of quality assessments have been based on comparisons of VGI datasets with other datasets, intrinsic assessments have recently gained attention [17]. In [20] it has been demonstrated that an estimation of the completeness of a dataset could be derived from an analysis of the community activity over time periods. This estimation classifies the mapping progress of a raster cell or region in three activity stages, namely "Start", "Growth" and "Saturation". The mapping activity itself (number of created features in a certain time period) is responsible for transitions between different stages. After the initial stage ("Start"), the "Growth" stage indicates that contributors have mapped the majority of the features during this period. A respective region proceeds to the final stage ("Saturation") when the number of created features per year drops under a pre-defined threshold (less than three percent of total count). Saturation means that most of the mapping, with respect to a selected feature type, has been completed. From a technical point of view, *Ac Create* and *Ac Delete* actions have to be analyzed in order to determine the increase/decrease of features within the raster cell and the given time period. Actions are filtered by *feature type* (e.g., street), a pre-selected *raster* (e.g., grid, hexagons or administrative boundaries) and *time period* (e.g., year 2014). For efficiently executing the analyses a separate analysis process for each raster cell is started. Therefore, the analysis task is well suited to be massively parallelized. The spatially indexed PBF files are used to read the data for one raster cell efficiently. As the same or nearby quadrants are analyzed in a consecutive way, the data in lower quadrants is kept in cache to avoid frequent file reading which speeds up the process (up to 25% time saving). With this measure, analyses for large regions (e.g., countries or continents) and several time periods could be completed within short time frames (between five seconds and four minutes for hexagons with a diameter of 20 km, duration depends on number of features within region). Also, continuous worldwide estimates (e.g., on a daily basis) are feasible. As an example, Figure 8 shows the activity stages for the feature type street in Europe by the end of 2014. While most parts of Europe still show low to high growth, the mapping progress is already saturated in Great Britain, Denmark, the Netherlands as well as some areas of Northern and Eastern Europe. The figure also reveals the highest feature growth in the eastern parts of Europe.

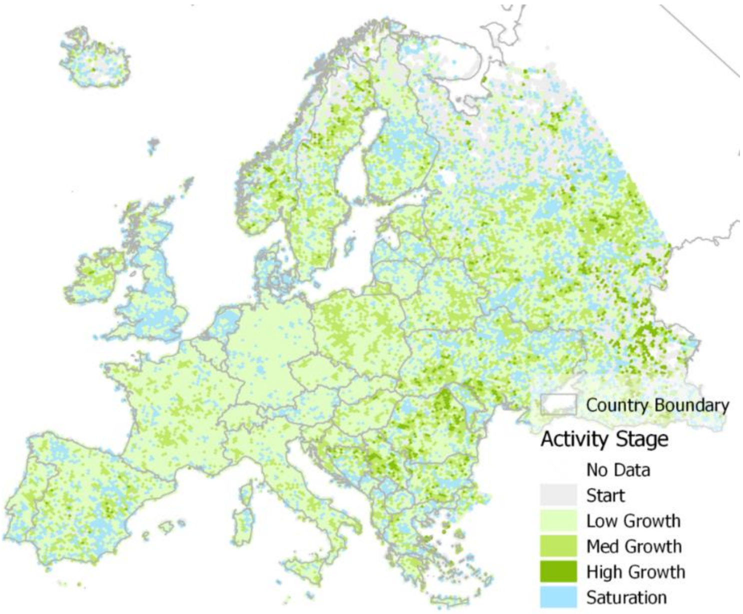
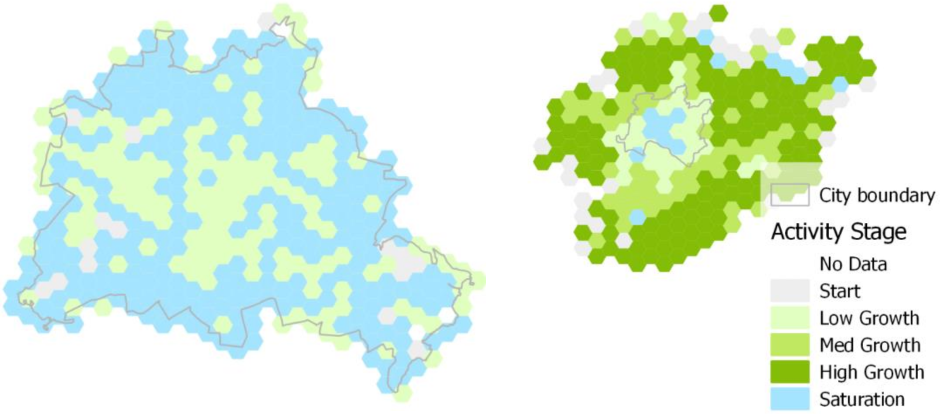


Figure 8. Map visualizing the activity stages for the feature class street in Europe by the end of the year 2014.

Figure 9 shows detailed maps of (a) Berlin and (b) Kathmandu illustrating activity stages by the end of 2015. As this analysis is based on the same dataset of 14 September 2015, the values of 2015 had to be extrapolated for the whole year (including the mapping activity after the earthquake disaster). Raster cell size has been set to a diameter of two kilometers compared with the previously used 10 km. The maps reveal some differences between both cities. While in Berlin the mapping progress is rather *saturated*, the mapping progress within the city boundaries of Kathmandu is in a phase between *low growth* and *saturation* since most streets have been mapped before 2015. However, the city's surrounding area features predominantly *medium* and *high growth* cells indicating that many streets in these regions have been mapped in 2015.



(**a**) (**b**)

Figure 9. Activity stages for the feature type street by the end of 2015 (extrapolated) for (**a**) Berlin and (**b**) Kathmandu.

5.3. Change Detection

A third interesting question is concerned with the temporal evolvement of geospatial vector datasets (especially but not only VGI datasets). Data analysts may be interested in feature changes occurring between two published versions of a dataset. The work in Rehrl *et al.* [22] proposes a method for the qualitative assessment of feature changes based on an analysis of actions. Again, the proposed framework and the following workflow are suitable for preparing the data. Operations of each feature within a selected region are read and aggregated to actions. All actions being executed within the analysis time period are classified per change type. While attribute and feature changes are derived directly from operations, geometry changes are identified by comparing two consecutive geometry versions which are assembled from the operations associated to the actions. Split and merged features should receive special importance for separating split- or merge-related create and delete operations from other create and delete operations.

For validating the framework, the change detection approach has been applied to activity data from the city of Berlin classifying all changes within three half-year long time periods between January 2014 and June 2015. Table 10 shows the number of changes for the feature type "street" per edit type for Berlin for three consecutive six-month periods. The numbers indicate a trend for a uniform development.

As proposed in previous work [22], a classification of changes according to five change types (*Feature Creation*, *Feature Deletion*, *Identity Change*, *Semantic Change* and *Feature Revision)* helps in separating relevant from irrelevant changes. Figure 10 shows a map of the city of Berlin highlighting changes by different change types. One prominent example for a changed identity happened with the Paul-Löbe-Haus in Berlin (large red polygons in Figure 10). As the detailed map of Berlin's city center shows, this building has been moved significantly, which resulted in a "changed identity" classification. As this has been obviously an erroneous mapping (since the building has not moved in reality), the building has been "virtually" moved back to its original location two days later. This example is well-suited proof for the necessity of change detection as well as qualitative classification.

Table 10. Number of street changes per type in Berlin for three half-year periods.

Edit Type	January–June 2014	July–December 2014	January–June 2015
Street Within Buffer	1.814	1.914	1.692
Geometry Split	541	765	873
Street Type Modified	584	302	516
Create Geometry	240	271	228
Street Shortened	181	224	182
Street Name Modified	207	224	169
Geometry Merge	132	83	100
Street Max Speed Modified	138	64	76
Delete Geometry	44	68	64
Street Lengthened	47	55	49
Street Crosses Buffer	68	48	29
Street Disjoint	5	2	3
Street Ref Modified	7	2	1
Total	4.008	4.022	3.982

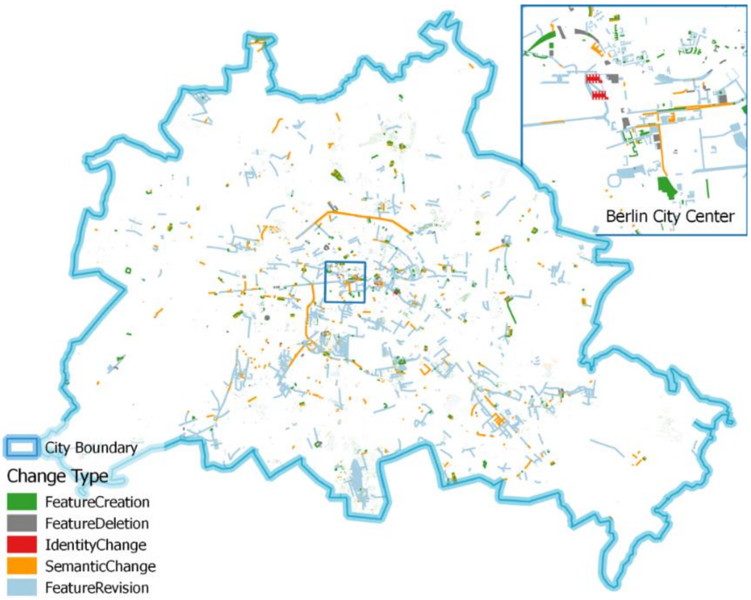


Figure 10. Change map of Berlin showing classified changes between January and June 2015.

Another example for the usefulness of change detection can be found in the city of Kathmandu before and after the earthquake (Figure 11). The change map indicates 411 new streets and 16,577 new buildings (green color) as well as 242 orange colored streets of change type "Semantic Change". Of those semantic changes, 127 reveal a modified street type, 28 reveal a modified street name, 30 streets have been realigned (new geometry crosses 20 meter buffer of old geometry), 21 streets have been lengthened and 36 streets have been shortened (more than the specified threshold of 20% total length). Other attribute-related changes are handled as feature revisions.

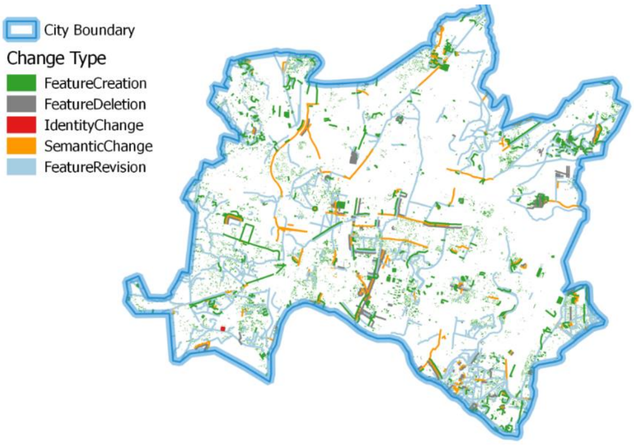


Figure 11. Change map of Kathmandu showing classified changes between January and June 2015.

6. Conclusions

In this work, a conceptual as well as technical framework for analyzing mapping activity in the context of VGI has been proposed. The framework builds upon the well-established theoretical model of *Activity Theory* as a conceptual frame for structuring mapping activity from a user perspective. This approach benefits from the widely accepted concepts *Activity*, *Action* and *Operation* having been established as key concepts for modelling user interaction in human–computer interaction (HCI) [26]. This work mainly adopts these concepts from a VGI contributor perspective. Using OSM as an example, the work clearly outlines the extraction of operations from OSM Full History data as well as the aggregation of operations to feature-related actions. Besides shaping the conceptual frame for activity modelling, the work shifts activity analyses from project-specific data models (e.g., OSM data model) onto the well-established feature model of the OGC Abstract Specification. This step has benefits in bridging the gap between community-driven data models and GIS-related data models, leading to a better comparability, reproducibility and interpretability of analyses results.

In addition to the conceptual frame, a technical framework for data processing has been proposed. This framework not only implements the proposed conceptual frame, but also allows for fast, worldwide data processing by using an optimized data structure for storing operations as well as an efficient spatial index structure based on quadtrees. The operation's format (PBF) has been proven to be a well-suited intermediate format for being used as a foundation for mapping activity analyses. Instead of starting each analysis from scratch by processing the OSM Full History File again, using the operation's format as a foundation has a clear advantage with respect to processing time. Most of the analyses for selected world regions run within minutes while the extraction of single regions from the OSM Full History typically needs daylong processing of the whole file. Another benefit of the technical framework is the numerous filter options, including spatial and temporal filters as well as filters for feature type or attribute level. Filter rules are simply set as input parameters before starting data processing. For instance, separating geometry-related from attribute-related updates may be achieved by simply changing the corresponding filter attributes.

The presented examples of mapping activity analyses have been chosen with consideration of different spatial as well as temporal criteria. Community activity over time periods has been proven to be a good estimate for completeness. With the proposed framework such estimates can be repeated in short time periods giving an indication for the overall mapping progress within a certain region. In case of special events such as natural disasters, the framework allows for fast estimation of the local mapping activity being useful for disaster response. Including daily snapshots in the analyses could help to tailor the mapping effort to certain regions being in need of additional mapping. Again, the overall progress of the mapping activity after the disaster may be estimated. Change detection is an interesting field of research which is not only for use in the context of VGI. Detecting changes between different snapshots of vector datasets is a valuable input for analytical or visual data quality assessment. The example of the erroneous OSM updates concerning the "Paul-Löbe-Haus" building in Berlin impressively demonstrates the usefulness of such analyses.

To conclude, the proposed framework provides a powerful tool to the VGI research community for numerous further analyses of mapping activities. The presented examples demonstrate first use cases while leaving numerous research questions for future work. As a future direction, the framework may be used as a foundation for empirical contributor studies, aiming at a better understanding of VGI community processes. A better understanding could help in predicting future developments of the dataset. A great help for the community would also be a near real-time monitoring of community activity as a live service. Raising awareness about mapping activities in the vicinity of community members could help for a better coordination of mapping efforts. Following this line of research, there is huge potential in developing an automated service for change detection. For instance, a change detection alert service with respect to the edits of a community member could help in monitoring changes to contributed features or features in the vicinity of one's home. Such a service would allow for an easy tracking of changes with the clear benefit of a better planning of mapping activities. Thus, one of the main future directions concerning the development and usage of the framework is not seen in additional analyses of historical data, but in live services helping VGI communities to better coordinate their mapping efforts. For leveraging the full potential of the framework and supporting a broader audience, the source code has been released as "VGI Analytics Framework" under the Apache License, Version 2.0 (https://github.com/SGroe/vgi-analytics-framework).

Acknowledgments: This work has been funded by the Austrian Ministry for Transport, Innovation and Technology (bmvit) as part of the Geoinformation Technology Lab Salzburg funding.

Author Contributions: Authorship has been strictly limited to researchers who have substantially contributed to the reported work. Karl Rehrl designed the overall methodology, interpreted results and contributed to outcome and conclusions. Simon Gröchenig implemented the software, conducted experiments and contributed results.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Goodchild, M.F. Citizens as sensors: The world of volunteered geography. *GeoJournal* **2007**, *69*, 211–221. [CrossRef]
2. Yasseri, T.; Sumi, R.; Kertész, J. Circadian patterns of Wikipedia editorial activity: A demographic analysis. *PLoS ONE* **2011**, *7*, e30091. [CrossRef] [PubMed]
3. Bryant, S.L.; Forte, A.; Bruckman, A. Becoming Wikipedian. In Proceedings of the 2005 International ACM SIGGROUP Conference on Supporting Group work—GROUP'05, Sanibel Island, FL, USA, 6–9 November 2005.
4. Mooney, P.; Rehrl, K.; Hochmair, H.H. Action and interaction in volunteered geographic information: A workshop review. *J. Locat. Based Serv.* **2013**, *7*, 291–311. [CrossRef]
5. Kottman, C.; Reed, C. *The OpenGIS Abstract Specification, Topic 5: Features*; Open Geospatial Consortium, Inc.: Wayland, MA, USA, 2009.
6. Budhathoki, N.R.; Raj, N.; Nedovic-Budic, Z. An Interdisciplinary frame for understanding volunteered geographic information. *Geomatica* **2010**, *64*, 313–320.
7. Nardi, B.A. *Activity Theory and Human-Computer Interaction*; The MIT Press: Cambridge, MA, USA, 1995.
8. Haklay, M.; Basiouka, S.; Antoniou, V.; Ather, A. How many volunteers does it take to map an area well? The validity of Linus' law to volunteered geographic information. *Cartogr. J.* **2010**, *47*, 315–322. [CrossRef]
9. 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. [CrossRef]
10. Neis, P.; Zielstra, D.; Zipf, A. Comparison of volunteered geographic information data contributions and community development for selected world regions. *Future Internet* **2013**, *5*, 282–300. [CrossRef]
11. Arsanjani, J.J.; Helbich, M.; Bakillah, M.; Loos, L. The emergence and evolution of OpenStreetMap: A cellular automata approach. *Int. J. Digit. Earth* **2013**, *8*, 1–15. [CrossRef]
12. Mooney, P.; Corcoran, P. Characteristics of heavily edited objects in OpenStreetMap. *Future Internet* **2012**, *4*, 285–305. [CrossRef]

13. Mooney, P.; Corcoran, P. Analysis of interaction and co-editing patterns amongst OpenStreetMap contributors. *Trans. GIS* **2013**, *18*, 633–659. [CrossRef]
14. Roick, O.; Hagenauer, J.; Zipf, A. OSMatrix-grid-based analysis and visualization of OpenStreetMap. In *State of the Map Europe*; Technical University of Vienna: Vienna, Austria, 2011; pp. 44–54.
15. Roick, O.; Loos, L.; Zipf, A. A Technical framework for visualizing spatio-temporal quality metrics of volunteered geographic information. In Proceedings of Geoinformatik 2012, Hong Kong, China, 15-17 June 2012; pp. 263–270.
16. Barron, C.; Neis, P.; Zipf, A. iOSMAnalyzer—Ein umfassendes Werkzeug für intrinsische OSM-Qualitätsuntersuchungen. In *AGIT 2013—Symposium und Fachmesse Angewandte Geoinformatik*; Wichmann: Salzburg, Austria, 2013; pp. 142–151.
17. Barron, C.; Neis, P.; Zipf, A. A comprehensive framework for intrinsic OpenStreetMap quality analysis. *Trans. GIS* **2014**, *18*, 877–895. [CrossRef]
18. Rehrl, K.; Gröchenig, S.; Hochmair, H.H.; Leitinger, S.; Steinmann, R.; Wagner, A. A conceptual model for analyzing contribution patterns in the context of VGI. In *Progress in Location-Based Services, Lecture Notes in Geoinformation and Cartography*; Krisp, J., Ed.; Springer-Verlag: Berlin, Germany, 2013; pp. 373–388.
19. Steinmann, R.; Brunauer, R.; Gröchenig, S.; Rehrl, K. Contribution profiles of voluntary mappers in OpenStreetMap. In Proceedings of the 1st International Workshop on Action and Interaction in Volunteered Geographic Information, Leuven, Belgium, 14–17 May 2013.
20. Gröchenig, S.; Brunauer, R.; Rehrl, K. Estimating completeness of VGI datasets by analyzing community activity over time periods. In *Lecture Notes in Geoinformation and Cartography, Connecting a Digital Europe through Location and Place*; Huerta, J., Schade, S., Granel, C., Eds.; Springer International Publishing: Cham, Switzerland, 2014; pp. 3–18.
21. Gröchenig, S.; Brunauer, R.; Rehrl, K. Digging into the history of VGI data-sets: Results from a worldwide study on OpenStreetMap mapping activity. *J. Locat. Based Serv.* **2014**, *8*, 198–210. [CrossRef]
22. Rehrl, K.; Brunauer, R.; Gröchenig, S. Towards a qualitative assessment of changes in geographic vector datasets. In *AGILE 2015: Geographic Information Science as an Enabler of Smarter Cities and Communities*; Bacao, F., Santos, M.Y., Pahino, M., Eds.; Springer International Publishing: Cham, Switzerland, 2015.
23. Engeström, Y. *Learning by Expanding: An Activity-Theoretical Approach to Developmental Research*, 2nd ed.; Cambridge University Press: Cambridge, 2015.
24. Kaptelinin, V.; Nardi, B.A.; Macaulay, C. Methods & tools: The activity checklist: A tool for representing the "space" of context. *Interactions* **1999**, *6*, 27–39.
25. Kuutti, K. Activity theory as a potential framework for human computer interaction research. In *Context and Consciousness: Activity Theory and Human-Computer Interaction*; Nardi, B.A., Ed.; The MIT Press: Cambridge, MA, USA, 1996; pp. 17–44.
26. Kaptelinin, V.; Nardi, B. *Acting with Technology: Activity Theory and Interaction Design*; The MIT Press: Cambridge, MA, USA, 2006.
27. Timpf, S. Geographic task models for geographic information processing. In Proceedings of 2001 Meeting on Fundamental Questions in Geographic Information Science; Manchester, UK, 1–2 July; pp. 217–229.
28. James, M. *Managing the Data-based Environment*; Prentice Hall: Upper Saddle River, NJ, USA, 1983.
29. Kottman, C. *The OpenGIS Abstract Specification, Topic 8: Relationships Between Features*; Open Geospatial Consortium, Inc.: Wayland, MA, USA, 1999.
30. Ramm, F.; Topf, J. *OpenStreetMap*, 3rd ed.; Lehmanns Media: Berlin, Germany, 2010.
31. Gröchenig, S. Using Spatial and Temporal Editing Patterns for Evaluation of Open Street Map Data. Master's Thesis, Carinthia University of Applied Sciences, Villach, Austria, 2012.
32. Samet, H. The quadtree and related hierarchical data structures. *ACM Comput. Surv.* **1984**, *16*, 187–260. [CrossRef]
33. Mooney, P.; Corcoran, P. How social is OpenStreetMap? In Proceedings of the 15th Association of Geographic Information Laboratories for Europe International Conference on Geographic Information Science, Avignon, France, 24–27 April 2012.