Proceedings

Clodoveu A. Davis Jr. and Karine Reis Ferreira (editors)

Dados Internacionais de Catalogação na Publicação

SI57a Simpósio Brasileiro de Geoinformática (11. : 2014: Campos do Jordão,SP)

Anais do 15º Simpósio Brasileiro de Geoinformática, Campos do Jordão, SP, 30 de novembro a 3 de dezembro de 2014. / editado por Clodoveu A. Davis Jr. (UFMG), Karine R. Ferreira (INPE). – São José dos Campos,
SP: MCTI/INPE, 2014.
Pendrive + On-line
ISSN 2179-4820

1. Geoinformação. 2.Bancos de dados espaciais. 3.Análise Espacial. 4. Sistemas de Informação Geográfica (SIG). 5.Dados espaço-temporais. I. Davis Jr., C.A. II.Ferreira, K.R. III. Título.

CDU: 681.3.06

Preface

This volume contains papers accepted and presented at the XV Brazilian Symposium on Geoinformatics, Geoinfo 2014, held in Campos do Jordão, Brazil, from November 30 to December 3, 2014. The GeoInfo conference series, inaugurated in 1999, comes to its fifteenth edition in 2014.

GeoInfo 2014 celebrates the 15th year of the event that is the most important academic meeting on geoinformatics and related subjects in Brazil. Altogether, the 15 editions of GeoInfo so far have produced 260 articles, involving nearly 600 authors – 49 of which have their first GeoInfo paper published in the 2014 proceedings. A total of 31 internationally acclaimed keynote speakers have addressed GeoInfo audiences since 1999; a full list can be obtained at GeoInfo's Web site, http://www.geoinfo.info, along with every published paper and details on every edition.

As usual, GeoInfo brings together researchers, students and participants from several Brazilian states and from abroad. The Program committee selected 20 papers, submitted by 69 distinct authors from 20 distinct Brazilian academic and institutions and research centers, from 8 Brazilian states and from Italy, France and the United States. Most contributions have been presented as full papers, but both full and short papers are assigned the same amount of time for oral presentation at the event. Short papers, which usually reflect ongoing work, receive a larger time-share for questions and discussions, so that the authors can benefit from the qualified symposium attendance. The symposium included special keynote presentations by Max Egenhofer, revisiting his first keynote talk at the first GeoInfo in 1999, and Paul Brown. Such highly qualified keynote speakers maintain GeoInfo's tradition of attracting some of the most prominent researchers in the world to productively interact with our community, thus generating all kinds of interesting exchanges and discussions, not only during presentation sessions, but also along break times and social events in the warm and cozy atmosphere of Campos do Jordão.

We would like to thank all Program Committee members and additional reviewers, listed below, whose work was essential to ensure the quality of every accepted paper. At least three specialists from the PC contributed with their review to each paper submitted to GeoInfo.

Our warmest thanks to the many people involved in the organization and execution of the symposium, particularly INPE's invaluable support team: Daniela Seki, Janete da Cunha, Denise Nascimento and Gláucia Pereira da Silva.

Finally, we would like to thank GeoInfo's supporters, the São Paulo Research Foundation (FAPESP), the Society of Latin American Remote Sensing Specialists (SELPER-Brasil), and Boeing Research & Technology, identified at the symposium's Web site and in this volume. The Brazilian National Institute of Space Research (Instituto Nacional de Pesquisas Espaciais, INPE) has provided once again much of the energy and commitment required to bring together this research community, now as in the past, and continues to perform this role through their numerous research and related activities.

Belo Horizonte and São José dos Campos, Brazil, November 2014.

Clodoveu A. Davis Jr.
Program Committee Chair

Karine Reis Ferreira
General Chair

Conference Committee

General Chair

Karine Reis Ferreira
National Institute for Space Research, INPE

Program Chair

Clodoveu A. Davis Jr.
Universidade Federal de Minas Gerais, UFMG

Local Organization

Daniela Seki
INPE

Gláucia Pereira da Silva
INPE

Janete da Cunha
INPE

Denise Nascimento
INPE

Support

FAPESP – São Paulo Research Institute
SELPER-Brasil – Associação de Especialistas Latinoamericanos em Sensoriamento Remoto
Boeing Research & Technology Brazil

Program Committee

Clodoveu A. Davis (Program Chair), Universidade Federal de Minas Gerais (UFMG), Brazil
Karine R. Ferreira (General Chair), National Institute for Space Research (INPE), Brazil

Ana Paula Afonso, Universidade de Lisboa, Portugal
André Santanchè, UNICAMP, Brazil
Antônio Miguel V. Monteiro, INPE, Brazil
Armanda Rodrigues, Universidade Nova de Lisboa, Portugal
Bart Kuijpers, Hasselt University, Belgium
Carla Macario, Embrapa, Brazil
Cláudio Baptista, UFCG, Brazil
Dieter Pfoser, George Mason University, USA
Edzer Pebesma, University of Munster, Germany
Flávia Feitosa, INPE, Brazil
Frederico Fonseca, The Pennsylvania State University, USA
Gilberto Câmara, INPE, Brazil
Gilberto Ribeiro de Queiroz, INPE, Brazil
Helen Couclelis, University of California, Santa Barbara, USA
João Pedro C. Cordeiro, INPE, Brazil
Jorge Campos, UNIFACS, Brazil
Jugurta Lisboa Filho, UFV, Brazil
Julio César L. D'Alge, INPE, Brazil
Laercio Namikawa, INPE, Brazil
Leila M. G. Fonseca, INPE, Brazil
Leonardo G. Azevedo, IBM Research Brazil (IBM) - PPGI/DIA (UNIRIO), Brazil
Lúbia Vinhas, INPE, Brazil
Luis Otávio Alvares, UFSC, Brazil
Marcelino P. S. Silva, UERN, Brazil
Marcelo Tílio M. de Carvalho, PUC Rio, Brazil
Marco Antônio Casanova, PUC Rio, Brazil
Marcus Vinicius A. Andrade, UFV, Brazil
Maria Isabel S. Escada, INPE, Brazil
Mário J. Gaspar da Silva, Universidade de Lisboa, Portugal
Matt Duckham, University of Melbourne, Australia
Pedro R. Andrade, INPE, Brazil
Raul Q. Feitosa, Pontifícia Universidade Católica do Rio de Janeiro, Brazil
Renato Fileto, UFSC, Brazil
Ricardo R. Ciferri, UFSCAR, Brazil
Ricardo S. Torres, UNICAMP, Brazil
Sergio D. Faria, UFMG, Brazil
Sergio Rosim, INPE, Brazil
Silvana Amaral, INPE, Brazil
Stephan Winter, University of Melbourne, Australia
Thales Sehn Korting, INPE, Brazil
Tiago G. S. Carneiro, UFOP, Brazil
Vagner Sacramento, UFG, Brazil
Valéria G. Soares, UFPB, Brazil
Valéria C.Times, UFPE, Brazil
Vania Bogorny, UFSC, Brazil
W. Randolph Franklin, Rensselaer Polytechnic Institute, USA
Werner Kuhn, University of California, Santa Barbara, USA

External Reviewers

André Salvaro Furtado
Cédric Grueau
Fernando José Braz
Lucas Vegi
Wagner Souza

Contents

Analysis of Spatiotemporal Inconsistencies of Trajectories with Planned and Reported Tasks

Felipe Pinto da Silva, Renato Fileto

PPGCC/INE, Federal University of Santa Catarina (UFSC)
PO BOX 476, 88040-900, Florianópolis-SC, BRAZIL

fpsilva98@gmail.com, r.fileto@ufsc.br

Abstract. *Trajectories automatically collected by sensors enable tracking moving objects that have to perform tasks in the geographic space. However, most existing methods that use such trajectories for analyzing moving objects behaviors do not use other relevant data sources. This paper proposes a method to detect and classify spatiotemporal inconsistencies of trajectories with both planned tasks and reported tasks. The classified spatiotemporal inconsistencies returned by the proposed method help to investigate possible misbehaviors of moving objects. This method has been implemented and evaluated with real data from a water supply company. It has helped to detect and investigate a variety of inconsistencies in these data.*

1. Introduction

Nowadays, a variety of mobile devices such as cell phones equipped with geographic positioning technologies (e.g., GPS, GSM) allow the automatic gathering of raw trajectories, i.e., temporally ordered sequences of spatiotemporal positions occupied by moving objects. Trajectories have more precise and detailed spatiotemporal information than traditional travel diaries, for example. Thus, they are a suitable source of information to analyze moving objects' behaviors.

Trajectories processing methods to investigate moving objects' behaviors while these objects (are supposed to) perform tasks in the geographical space have recently attracted the attention of some research groups. These methods have many applications, ranging from the management of itinerant teams doing public or commercial services (e.g., public works, maintenance in place, domiciliary inspection, in-home caregiving) to delivery services (e.g., conventional mail, pizza delivery). However, the analysis of data from other sources, such as tasks planning and travel diaries, along with trajectories allow the extraction of further relevant information about the tasks (e.g., type, goal), and other activities that may influence tasks execution performance (e.g., idle time).

This paper proposes a computational method to automatically detect and classify spatiotemporal inconsistencies of moving objects' trajectories with their planned tasks and tasks that they report to have done in specific spatiotemporal points. The inputs of the proposed method are: (i) raw trajectories of moving objects; (ii) the execution place and the expected duration of each planned task; and (iii) reports made by the moving objects themselves, with (possibly inaccurate) indications of the ending time, duration, and geographic coordinates of each executed task . The proposed method confronts the data from these three sources, and classifies each found inconsistency. The returned

spatiotemporal inconsistencies among the data of distinct nature and sources support the investigation of operational problems and possible misbehaviors of moving objects. The proposed method has been implemented in a prototype that uses state-of-the-art methods for spatiotemporal data processing. Experiments with real data about the activities of maintenance teams of a water supply company have shown that the proposed method is able to detect and help to investigate a wide variety of inconsistencies.

The rest of this paper is structured as follows. Section 2 provides the foundations necessary to understand the proposal. Section 3 describes the proposed method in detail. Section 4 reports experimental results. Section 5 reports some related work, and Section 6 concludes the paper, by enumerating contributions and themes for future work.

2. Foundations

The inputs of the proposed method are trajectories, planned tasks, and reported tasks. They are formally described in the following.

2.1. Trajectories

A raw trajectory is a sequence of spatiotemporal positions of a moving object obtained from some sensor (GPS, GSM, RFID, cameras). Definition 2.1 formalizes this concept.

Definition 2.1. *A **raw trajectory*** $\tau = (idMO, idTraj, (x_1, y_1, t_1), \ldots, (x_n, y_n, t_n))$ *is a temporally ordered sequence of spatiotemporal positions of a moving object, where:*

- $idMO$ *is the identifier of the moving object;*
- $idTraj$ *is the unique identifier of the trajectory* τ*;*
- *each* (x_i, y_i, t_i) *is a spatiotemporal position (point), representing that the object is at the spatial coordinates* (x_i, y_i) *in the instant* $t_i (1 \leq i \leq n)$*; and*
- $n > 0$ *is the number of spatiotemporal positions of* τ*.*

Raw trajectories must be preprocessed to eliminate inaccuracies caused by limitations of the sensing devices and problems of the trajectory gathering process [Yan *et al*. 2013]. Then, their filtered and possibly adjusted spatiotemporal positions can be structured in episodes [Mountain and Raper 2001], described by Definition 2.2.

Definition 2.2. Given a trajectory $\tau = (idMO, idTraj, (x_1, y_1, t_1), \ldots, (x_n, y_n, t_n))$, *an **episode*** $E = (idTraj, idE, (x_{start}, y_{start}, t_{start}) \ldots (x_{end}, y_{end}, t_{end}))$ *is a sub-sequence of* τ $(1 \leq start \leq end \leq n)$ *that satisfies a given predicate, and is maximal in terms of its number of spatiotemporal points, where:*

- $idTraj$ *is the trajectory identifier;*
- idE *is the episode identifier;*
- t_{start} *is the instant of the first spatiotemporal point of the episode; and*
- t_{end} *is the instant of the last spatiotemporal point of the episode.*

Episodes can be moves or stops, for example. The positions constituting an episode such as a stop can be determined by distinct predicates (e.g., being inside a place, having speed below a given threshold). [Spaccapietra *et al.* 2008] introduced the trajectory model based on intertwined stops and moves, and defines a semantic trajectory as a sequence of relevant locations visited by the moving object. Structured trajectories are time ordered sequences of episodes [Yan *et al*. 2013] (Definition 2.3).

Definition 2.3. *A **structured trajectory** is a timely ordered sequence of temporally disjoint episodes* $T_s = \{E_1, E_2, \ldots E_m\}$ $(m > 0)$.

Comprehensive reviews about concepts and techniques related with trajectories data processing (e.g., detecting relevant episodes and behaviors) can be found in [Parent *et al*. 2013] and [Pelekis and Theodoridis 2014]. A method for building structured trajectories by extracting stops on a given set o places is proposed in [Alvares *et al*. 2007]. This method is called IB-SMoT (Intersection-Based Stops and Moves of Trajectories). CB-SMoT (Cluster-Based SMoT) is a method for structuring trajectories based on speed [Palma *et al*. 2008]. It is able to identify stops by clustering adjacent positions in which the moving object is stationary or moves slowly.

In this paper we consider that a stop is necessary for a moving object to perform a task. This assumption is also used in [Huang *et al*. 2010], [Furletti *et al*. 2013] and [Clark & Doherty 2008]. Then, as we are interested in analyzing inconsistencies between stops, tasks planning and reported tasks, we use CB-SMoT to detect stops in any location, even locations where there is no reported or planned task.

2.2. Tasks

According to Huang, a task has location, initial instant, duration, and goal [Huang *et al*. 2010]. However, the proposed method does not impose rigid schedule. i.e., doesn't matter when a task should start, but the minimum and maximum expected duration.

Definition 2.4. *A **planned task** is a tuple* φ = (idMO,idTask,(x,y), minExpDuration,maxExpDuration,requestDate,maxCompletionDate), where:

- idMO is the identifier of moving object;
- idTask is the identifier of planned task;
- (x,y) are the spatial coordinates where the planned task should be performed;
- minExpDuration ≤ maxExpDuration are respectively the minimum and the maximum expected duration of the planned task;
- requestDate ≤ maxCompletionDate are respectively the request date and the maximum allowed completion date of the planned task execution.

Clark e Doherty [Clark & Doherty 2008] added feedback from users to analyze rescheduling tasks. These feedbacks only confirms the start execution and finish execution. Our method introduces the concept of reports made by the moving objects.

Definition 2.5. *A **reported task** is a tuple* ρ = (idMO,idTask,startTime,endTime, (x,y)), where

- idMO is the identifier of moving object;
- idTask is the identifier of planned and reported task;
- startTime is the starting time of the task execution;
- endTime is the end time of the task execution;
- (x,y) are the spatial coordinates from which the report was sent.

3. Analysis of Spatiotemporal Inconsistencies

This section describes the proposed method to analyze spatiotemporal inconsistencies of trajectories with planned and reported tasks. Section 3.1 presents an overview of the

method, while Sections 3.2 and 3.3 detail the method steps of inconsistencies detection and inconsistencies classification, respectively.

3.1. Overview

Figure 1 summarizes the proposed method. Its inputs are T, P, and R. T is a set of raw trajectories (Definition 2.1). P is a set of planned tasks (Definition 2.4). R is a set of reported tasks (Definition 2.5). Step 1 (Preprocessing and Extraction of Stop Episodes) starts by preprocessing each trajectory in T to generate a set of structured trajectories T_S (Definition 2.3). Each structured trajectory in T_S is a sequence of stops and moves (Definition 2.2). These episodes are extracted by using a cluster-based method such as CB-SMoT [Palma *et al*. 2008]. For simplicity and efficiency of the next step, the location of each stop is approximated by a single point, such as its centroid. The stops of all the structured trajectories of T_S are inserted in the relation S that is used as one of the inputs for step 2 (Detection of Inconsistencies).

Step 2 executes spatiotemporal SQL queries on sets of trajectory stops, planned tasks, and report tasks. The third and last step (Classification of Inconsistencies) helps to explain the inconsistencies found. Steps 2 and 3 are described in the following. The outputs of the proposed method are classified spatiotemporal inconsistencies.

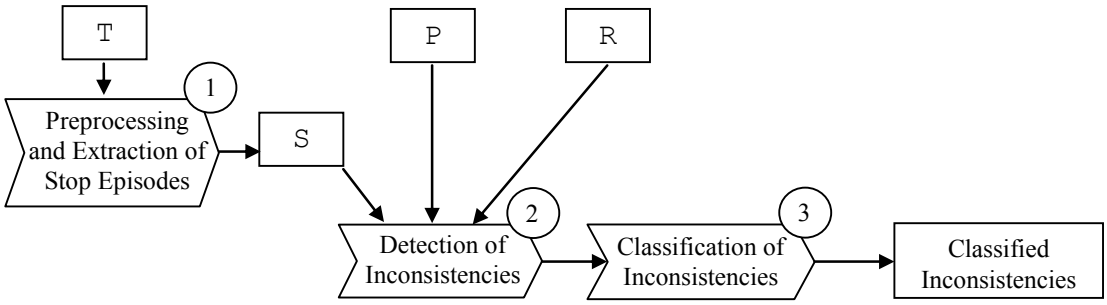


Figure 1. Data processing steps of the proposed method

3.2. Detection of Inconsistencies

The detection of inconsistencies is done by exploiting the relational database view whose query expression is presented in Figure 2. This query applies successive outer joins on the tables: planned tasks (P), reported tasks (R), and the stop episodes of the structured trajectories (S). All these datasets refer to the same set of moving objects during a certain time period (e.g., a month). R and S take part in more than one join in the query, with distinct join conditions. It is important to distinguish the attribute values (and *NULL* values) resulting from each outer join to have all the needed information to detect different kinds of inconsistencies in the final result (Res). Thus, we use R' and S', which are just the projections of the attributes of R and S, respectively, necessary for the respective join conditions. In other words:

$$R' = \pi_{idMO,startTime,endTime,x,y}(R)$$

$$S' = \pi_{idMO,startTime,endTime,x_c,y_c}(S)$$

The attributes of P, R, and S are as stated in Definitions 2.4, 2.5, and 2.2, respectively. The coordinates ($S.x_c$, $S.y_c$) can refer to the centroid of each stop episode,

for example. These coordinates are used instead of the whole subsequence of spatiotemporal points of each stop to represent it, for simplicity and efficiency.

The first operation of the query presented in Figure 2 (a) is a full outer join between `P` and `R`. The join condition is a conjunction of equalities of the attributes `idMO` and `idTask` of both tables(natural join condition), with the time compatibility of the planned task with the reported task (i.e., `P.requestDate` and `P.maxCompletionDate` less or equal `R.startTime` and `R.endTime`, respectively) This join enables the identification of planned tasks that do not match any reported task (i.e., planned tasks that were not realized or whose execution was not reported), and vice versa (i.e., reported tasks that do not correspond to any planned task).

The other outer joins in the query presented in Figure 2 (b, c, and d) have as join conditions the conjunction proximity of the coordinates of the respective spatiotemporal data (e.g., `distance((P.x,P.y),(R'.x,R'.y)) ≤ SThreshold_PR`), with moving object and time compatibility conditions. The spatial thresholds `SThreshold_PR`, `SThreshold_PT` and `SThreshold_RT` can be adjusted according to the dataset properties and application goals. Thus, one spatial point is considered close to another one if the distance between them is smaller than the respective spatial threshold.

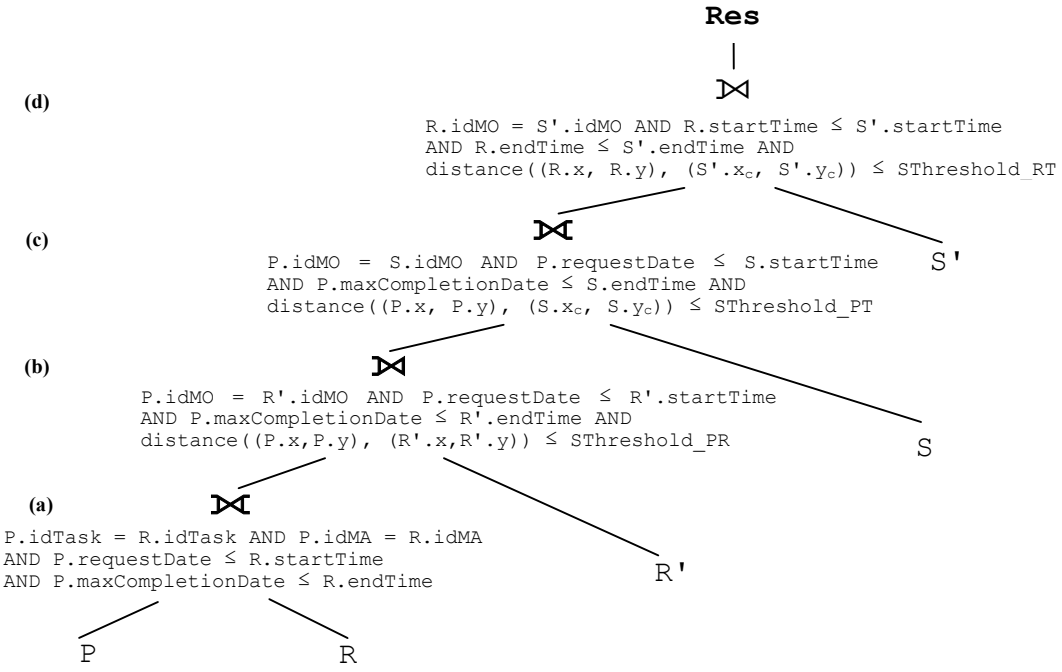


Figure 2. Definition of the view used for inconsistencies detection

The proposed method detects inconsistencies of each kind by posing a distinct query on the view `Res`. These queries filter tuples of `Res` presenting some partial matchings among planned tasks, reported tasks, and stop episodes referring to the same moving object. The planned task and the reported task must refer to the same `idTask`, as stated in Figure 2, to avoid combinatorial explosion. The considered partial matchings are detected by selection based on conjunctive conditions that exploit the values or values absence (*NULL* value) of some attributes of `P`, `R`, `R'`, `S`, and `S'`.

For example, if a reported task position is too far away from its corresponding planned task position, then `P.idTask` and `R.idTask` are not null and have the same value in `Res`, but `R'.idReport` is null, because `R'` has been joined with `P` by proximity to produce `Res`. Temporal inconsistencies, on the other hand, are detected by checking the temporal attributes in `Res`. For example, if a given moving object reports a task lasting longer than its maximum expected duration, then:

$$(R.endTime - R.startTime) > P.maxExpDuration$$

Figure 3 exemplifies the detection of planned tasks matching a reported task and a trajectory stop, but whose matching reported task duration is shorter than the minimum expected time to perform that task.

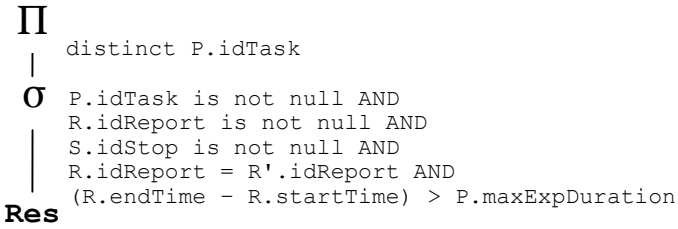


Figure 3. Retrieving inconsistencies between planned duration and reported duration

Analogous queries can extract other kinds of inconsistencies by changing the restrictions on the attributes of `Res` used to filter its tuples. Figure 4 presents the extraction of planned and reported tasks whose report location is far from planned location. Moreover, the reported duration is greater than the maximum expected duration, and the moving object has not stopped close to the planned location.

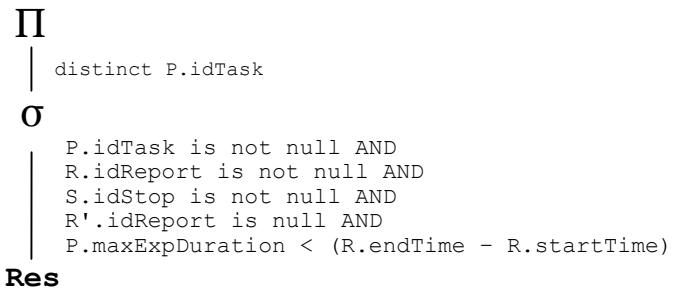


Figure 4. Retrieval of spatiotemporal inconsistencies on tasks location and duration

Figure 5 illustrates an inconsistency retrieved by the query presented in Figure 4. Let $\varphi \in P$ be a planned task, $\varphi.minExpDuration = 20$, and $\varphi.maxExpDuration = 30$, both in minutes. This task should be performed inside the region delimited by the blue circle labeled A. The red circles represent the centroids of two stops (s_1 , $s_2 \in S$) of the moving object assigned to execute φ. Consider $(s_1.endTime - s_1.startTime) = 20$ and $(s_2.endTime - s_2.startTime) = 10$. If the reported task $\rho \in R$ was done during the stop s_2, is less than `SThreshold_PT` meters away from s_2, and has the same `idTask` as that of φ, then there is a spatial inconsistency between φ and ρ, and a duration inconsistency between ρ and s_2.

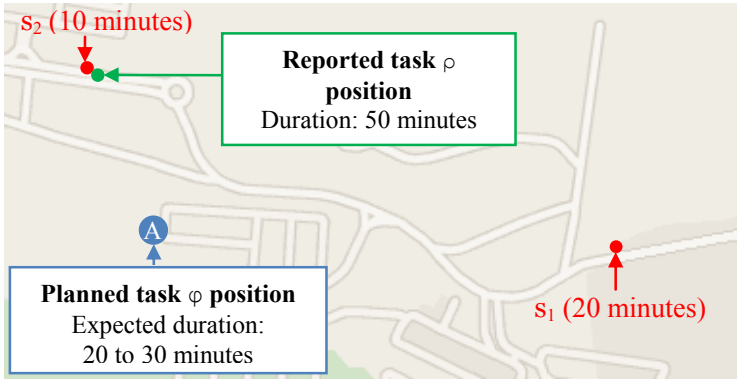


Figure 5. Detection of inconsistency in figure 4

3.3. Classification of Inconsistencies

The inconsistencies detected by the previous task are classified according with a conjunction of conditions that can hold or not on tuples of the view `Res`. These conditions can be disposed in a truth table for systematic assessment. They are expressed by possible *NULL* values on the attributes of planed tasks, reported tasks or trajectory episodes (φ, ρ, ρ', s, s'), which indicate lack of matching components for natural or similarity joins of the relational algebra expression that defines `Res` (Figure 2). The following temporal conditions are also considered in the truth table:

c_1: ρ.duration ≤ φ.maxExpDuration

c_2: ρ.duration ≥ φ.minExpDuration

c_3: s.duration ≤ φ.maxExpDuration

c_4: s.duration ≥ φ.minExpDuration

The duration of a reported task ρ or stop s can be determined as follows:

ρ.duration = (ρ.endTime − ρ.startTime)

s.duration = (s.endTime − s.startTime)

Thus, the truth table used to classify inconsistencies has 2^7 combinations of Boolean values, indicating if the attributes of P, R, R', S, and S' exist (are not *NULL*), and if the temporal conditions c_1, c_2, c_3, and c_4 are satisfied. The number of lines of the fact table can be reduced by eliminating combinations of Boolean values that do not make sense (e.g., if attributes of P or R are *NULL* then the conditions c_1 and c_2 cannot be evaluated), and by considering only combinations of conditions that express relevant classes of inconsistencies (e.g., for a particular company).

Table 1 presents 10 classes of inconsistencies considered relevant by the water supply company of our case study to investigate possible misbehavior of its maintenance teams. The symbols ∃ and ∄ indicate if the attributes of P, R, R', S, and S' are present or *NULL*, respectively, in the view `Res`. The columns c_1, c_2, c_3 and c_4 are checked (✓) if the respective temporal condition is satisfied, unchecked (✗) if they are satisfied, and marked as not possible to assess (-) otherwise. The class 1 refers to no inconsistency, i.e., the reported task and the stop positions are sufficiently close to that of the planned task, and their durations are within the expected time limits of the planned task. The

remaining classes presented in Table 1 refer to inconsistencies whose explanations appear in the last column.

Table 1. Some relevant classes of inconsistencies and their explanations

C	P.*	R.*	R'.*	S.*	S'.*	c_1	c_2	c_3	c_4	Explanation
1	∃	∃	∃	∃	∃	✓	✓	✓	✓	No inconsistency
2	∃	∃	∃	∃	∃	✓	×	✓	✓	Short reported task
3	∃	∃	∃	∃	∃	×	✓	✓	✓	Prolonged reported task
4	∃	∃	∄	∃	∃	×	✓	✓	×	Prolonged reported task, short stop, and wrong location of task report
5	∃	∃	∄	∄	∄	×	✓	-	-	Prolonged reported task, and planned task location not visited
6	∃	∄	∄	∃	∄	-	-	✓	×	Unreported task and planned location shortly visited
7	∃	∃	∄	∃	∄	×	✓	✓	✓	Prolonged reported task at planned location that is not visited.
8	∃	∄	∄	∄	∄	-	-	-	-	Task not performed
9	∄	∃	∄	∄	∄	-	-	-	-	Report of unknown task
10	∄	∄	∄	∃	∄	-	-	-	-	Idle stop

The inconsistency class 5 is detected by the query in Figure 4, and illustrated by task A of Figure 5. This inconsistency has no stops close to the planned task location. However, the reported task position is far away from its corresponding planned task position, and the reported duration is greater than the maximum expected duration. This characterizes a possible fraud.

4. Experiments

The proposed method has been implemented in Java, on top of a database managed with PostgreSQL 9.1.4 (64-bit) and PostGIS 2.0.3 r11132. The queries were performed in a server with an i7-2670QM 2.20GHz processor, 8Gb RAM, and 750Gb HD 7200 RPM.

We evaluated the implemented prototype of the proposed method on data of a water supply company. Several service requests must be served in a geographic region for each maintenance team, within given deadlines. Each team carries a mobile device running an application that collects its geographic position at every minute. Moreover, this application allows the team to report the execution of each task, supposedly from its execution location. The devices used by teams ensure 10 meters of accuracy. For these experiments, each maintenance team is considered a moving object, and each service request is considered a planned task. The spatiotemporal coordinates of raw trajectories and reported tasks were collected using the GPS sensors of the mobile devices carried by the maintenance teams. The planned tasks were extracted from a company's database, which have only the addresses where the tasks should be performed. Thus, we have used geocoding to infer the spatial coordinates of planned tasks.

The set of daily trajectories used in the experiments contains over 900 thousand spatiotemporal positions, of 11 maintenance teams, collected between January 2008 and September 2013. During this period those teams send 6,468 reported tasks, and there were 8,505 planned tasks for them.

The Google Geocoding API was used to infer the planned tasks' coordinates from the street addresses of the service requests. The CB-SMoT method [Palma *et al*. 2008] implemented in Weka-SMTP [Bogorny *et al*. 2011] was used to extract stops from raw trajectories with the following parameters: 300 seconds as minimum duration to consider a stop, and 0,3 m/s as maximum allowed speed in a stop. It has generated 8,779 stops episodes from the company database. The values of spatial thresholds specified in subsection 3.2 were: `SThreshold_PR` equals to 15 meters; `SThreshold_PT` equals to 20 meters; and `SThreshold_RT` equals to 15 meters.

4.2. Results

The proposed method detected 16,955 inconsistencies. Table 2 shows the total number of inconsistencies classified in each class described in Section 3.3. The 4 columns in the right of this table show the percentages of inconsistencies of each class (C) relative to: the number of inconsistencies found (%); the number of planned tasks (%P); the number of reported tasks (%R); and number of stop episodes (%S). Only 29 tasks showed no inconsistencies (class 1). Possibly, 24% of the tasks were not performed (class 8). Idle stops represent almost half of the inconsistencies found (class 10)

Table 2. Summary of the experimental results

C	Possible explanation	Count	%	%P	%R	%S
1	No inconsistency	29	0.17	0.34	0.45	0.33
2	Short reported task	160	0.96	1.88	2.47	1.82
3	Prolonged reported task	1	0.01	0.01	0.02	0.01
4	Prolonged reported task, short stop, and wrong location of task report	0	0	0	0	0
5	Prolonged reported task, and planned task location not visited	105	0.63	1.23	1.62	-
6	Unreported task and planned location shortly visited	19	0.11	0.22	-	0.22
7	Prolonged reported task at planned location that is not visited.	103	0.62	1.21	1.59	-
8	Task not performed	2,070	12.43	24.34	-	-
9	Report of unknown task	0	0	-	0	-
10	Idle stop	8,249	49.53	-	-	93.96
-	Other classes	6,219	37.34	-	-	-

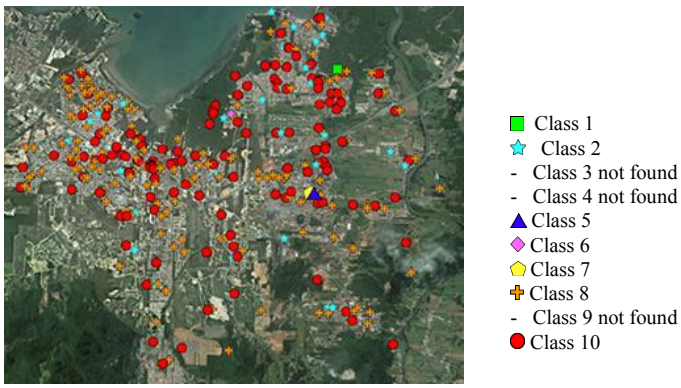


Figure 6. Geographical distribution of considered classes of inconsistencies

Figure 6 presents the geographical distribution of the inconsistencies of the classes presented in Table 2 found in a dataset collected between 1st and 30th June 2013. The cause of so many inconsistencies is currently under investigation yet.

These results illustrate how the proposed method allows the detection of a wider variety of inconsistencies than related work, by considering matching/non-matching trajectories, planned tasks, and reported tasks. In addition, our classification of the detected inconsistencies provides hints for their explanations.

5. Related Work

The work of [Raj *et al.* 2006] fuses data generated by GPS and other sensors to recognize activities performed by moving objects. The used sensors include a 3-axis accelerometer, microphones for recording speech and ambient sound, photo-transistors for measuring light conditions, temperature sensors, and barometric pressure sensors. The use of diverse sensors requires lots of configurations. In addition, the processing of multimodal data can be quite difficult and costly.

Trajectories are compared with a schedule of planned tasks in [Clark & Doherty 2008]. Their goal is to find rescheduled tasks by analyzing their execution locations (GPS coordinates), initial and final instants, planned tasks, and users' feedback. The findings are confirmed by the tasks executors via interviews. Their method does not use reported tasks, detects only temporal inconsistencies, and does not classify them.

The method for describing moving objects tasks and generating diaries introduced in [Huang *et al*. 2010] assumes that tasks along a trajectory occur at Points of Interest (PoIs), where the moving objects remain stationary for a while. According to them a task has a location (e.g., geographic coordinates), a starting time, duration, and a goal. Their method describes tasks according to categories of PoIs previously registered. Therefore, the quantity and the quality of the PoIs set are decisive for their method. They do not consider that a PoI can play different roles for different moving objects (e.g., a restaurant can be a lunch place for a moving object, and a job for another one).

The proposal of [Furletti *et al.* 2013] is similar to the one of [Huang *et al*. 2010], but instead of using PoIs previously registered, it uses PoIs from the Google Place API and OpenStreetMap. The type of a task is determined according to the type of location where the moving object stops to perform that task. Neither [Furletti *et al*. 2013] nor [Huang *et al*. 2010] use data of planned tasks or reported tasks.

Another novel method presented in [Zhenyu *et al*. 2012] generates a travel diary based on connections to Wi-Fi access points, and signal strength of mobile phones. Finally, the method proposed in [Yuan *et al*. 2013] constructs individual movement histories (similar to users' diaries) from a smart card transactions dataset. These works do not consider planned tasks. They rely just on the use of Wi-Fi access points and smart card transactions, respectively, to produce the travel diary information.

Table 3 compares related work. The columns "Trajectory", "Planned tasks" "User's reports", and "Other data sources" indicate if each method uses data from the respective sources. Notice that none of these related works classify spatiotemporal inconsistencies of trajectories with both planned tasks and reported tasks. For the best of our knowledge our proposal is the first one to consider all these data sources.

Consequently, our method is able to detect a wider variety of spatiotemporal inconsistencies than related proposals, and supports the investigation of the reasons for each detected inconsistency. In addition, our method does not impose a rigid schedule to start and finish tasks. It relies only on the set of tasks to be performed, and the expected position and duration of each task. Thus, it allows more flexibility for the moving objects to decide when to realize each task.

Table 3. Comparison to related works

Work	Trajectory	Planned tasks	User's reports	Other data sources	Classification of Inconsistencies
Clark & Doherty 2008	*Yes*	*Yes (Schedule)*	*No*	*No*	*No*
Furletti *et al.* 2013	*Yes*	*No*	*No*	*Google Place API and OpenStreetMap*	*No*
Huang *et al.* 2010	*Yes*	*No*	*No*	*Known PoI's*	*No*
Raj *et al.* 2006	*Yes*	*No*	*No*	*Sensors (accelerometer, microphones)*	*No*
Proposed method	***Yes***	***Yes***	***Yes***	***No***	***Yes***

6. Conclusions and Future Work

This paper introduced a computational method to detect and classify spatiotemporal inconsistencies of trajectories with planned and reported tasks. The main advantages of the proposed method are: (i) the use of 3 kinds of data (plans, tracks, and reports of moving objects performing tasks) to detect inconsistencies; (ii) capacity to detect a wider variety of spatiotemporal inconsistencies than state-of-the-art methods; (iii) higher flexibility for input data than related methods, by not using rigid schedules (with specific times for executing planned tasks); and (iv) several parameters to tune the method according to properties of the analyzed datasets. The returned spatiotemporal inconsistencies among data of distinct nature and sources help to investigate problems and possible misbehaviors of the moving objects (e.g., idle stops, frauds).

Future work comprises: (i) further validation of the proposed method, with distinct datasets and some ground true to asses results quality measures, such as precision and recall; (ii) evaluation of the performance and the scalability of the proposed method with bigger datasets; (iii) association of the geographical location of detected inconsistencies with geographic features of existing databases (e.g. OpenStreetMap) and linked data collections (e.g. LinkedGeoData) to better investigate the possible semantics of these inconsistencies; (iv) analysis of trajectory annotations and/or social media posts made by the moving objects to obtain additional information for explaining their behaviors (e.g., places and events visited, actions, and intentions).

Acknowledgments

Work supported by CNPq (grant 478634/2011-0), and FEESC. Thanks to the colleagues Andre S. Furtado, Juarez A. P. Sacenti, and Ricardo G. B. Nabo for their valuable help.

References

Alvares, L. O., Bogorny, V., Kuijpers, B., Macedo, J.A.F., Moelans, B. and Vaisman, A. (2007). *A model for enriching trajectories with semantic geographical information*. Proc. ACM-GIS, pp. 162–169, New York, NY, USA. ACM Press.

Bogorny, V., Avancini, H., de Paula, B. L., Kuplish, C.R., and Alvares, L. O. (2011). *Weka-STPM: a Software Architecture and Prototype for Semantic Trajectory Data Mining*. Transactions in GIS, 15:(2) 227-248

Clark, A. F., Doherty, S. T. (2008) *Use of GPS to automatically track activity re-scheduling decisions*. 8th Intl. Conf. on Survey Methods in Transport.

Furletti, B., Cintia, P., Renso, C., Spinsanti, L. (2013). *Inferring human activities from GPS tracks*. 2nd ACM SIGKDD Intl. Workshop on Urban Computing (UrbComp).

Huang, L., Li, Q., Yue, Y. (2010). *Activity identification from GPS trajectories using spatial temporal POI's attractiveness*. 2nd ACM SIGSPATIAL Intl. Workshop on Location Based Social Networks.

Mountain, D. and Raper, J. (2001). *Modelling human spatio-temporal behaviour: a challenge for location based services*. Intl. Conf. on Geocomputation, pages 24–26, Brisbane, Australia

Palma, A. T., Bogorny, V., Kuijpers, B., and Alvares, L. O. (2008). A clustering-based approach for discovering interesting places in trajectories. ACM SAC, pages 863–868, New York, NY, USA. ACM Press.

Parent, C., Spaccapietra, S., Renso, C., Andrienko, G., Andrienko, N., Bogorny, V., Damiani, M. L., Gkoulalas-divanis, A., Macedo, J., Pelekis, N., Theodoridis, Y., Yan, Z. (2013). *Semantic trajectories modeling and analysis*. ACM Comp. Surveys, 45(4).

Pelekis, N., Theodoridis, Y. (2014). *Mobility Data Management and Exploration*. Springer, ISBN 978-1-4939-0391-7, pp. 1-298

Spaccapietra, S., Parent, C., Damiani, M.L., Macêdo, J.A., Porto, F., Vangenot, C. (2008). *A conceptual view on trajectories*. Data Knowl. Eng. 65(1): 126-146

Raj, A., Subramanya A., Dieter F., Bilmes, J. (2006). *Rao-Blackwellized particle filters for recognizing activities and spatial context from wearable sensors*. 10th Intl. Symp. on Experimental Robotics. Springer Verlag.

Yan, Z., Chakraborty, D., Parent, C., Spaccapietra, S., Aberer, K. (2013) *Semantic trajectories: Mobility data computation and annotation*. ACM TIST 4(3): 49.

Yuan, N.J., Wang, Y., Zhang, F., XIE, X., Sun., G. (2013). *Reconstructing Individual Mobility from Smart Card Transactions: A Space Alignment Approach*. IEEE 13th Intl. Conference on Data Mining, 877-886

Zhenyu C., Shuangquan W., Yiqiang C., Zhongtang Z., Mu L., (2012) *InferLoc: Calibration Free Based Location Inference for Temporal and Spatial Fine-Granularity Magnitude*. IEEE 15th Intl. Conf. on Comp. Science and Engineering

Development of an application using a clustering algorithm for definition of collective transportation routes and times

Thiago C. Andrade[1], **Marconi de A. Pereira**[2], **Elizabeth F. Wanner**[1]

[1]DECOM - Centro Federal de Educação Tecnológica de Minas Gerais - Campus II - Brasil

[2]DTECH - Universidade Federal de São João del-Rei - Campus Alto Paraopeba - Brasil

Abstract. *This article presents the modeling, development and theoretical grounding for the development of an application based on the clustering algorithm DBSCAN, aiming to reduce the daily waste of time on the locomotion of a huge number of people to a common place. The clusters are created based on attributes, like the departure time of each person from its residence, the final destine and its both geographical locations (departure and arrival). People that compose the cluster are transported in a vehicle allocated according to the size of the cluster. A case study, with a specific group of people going to CEFET-MG Campus II, was conduct using a popular traffic simulator in order to measure the individual and the global time that people need to move, using the new transport arrangement. The simulation results were compared with a scenario where all people use public transportation. This comparison identified an average reduction of 45.80% in the time that people spent daily for their locomotion.*

1. Introduction

The traffic problems became more evident in the biggest metropolises in the last decades, due to the population growth. Thus, the Traffic Engineering started to have an important role in the traffic management of those cities, helping in the reduction of the problems arising from the huge number of vehicles on the streets, which contribute to the increase of locomotion time.

Based on this scenario, the natural tendency is that each one tries to solve out their own locomotion problem independently, choosing the solution which demands less time, also providing comfort. However, finding the best solution for each single person is not feasible. Thus, the most promising way to make traffic conditions better is to identify groups with similar locomotion patterns (origin, destination and time) and propose a viable way for them travel together. Those groups of people are known in literature as *traveling companions*. Some relevant works were presented, e.g. [Tang et al. 2012], [Tang et al. 2013], introducing methods of grouping people. However the challenge continues, since most of the solutions proposed are not practical.

The proposed problem can also be founded in literature related to school bus routing problem (SBRP) [Newton and Thomas 1969]. However, this class of problems considers many variables like bus garages and bus stop sequence, which increase the complexity of the problem [Park and Kim 2010]. On the other hand, it does not consider the possibility of using small vehicles, like cars, where a driver gives a ride to some neighbors.

Using the concepts of clusterization, the latitude, longitude and locomotion time of the individuals can be collected and processed to group people that have similar loco-

motion time and also similar origin and destination. Then, it is possible to find routes that can transport these groups of people efficiently.

Many works have been proposed, related to spatio-temporal clusterization, proposing improvements in the traffic, either through ride, new public transportation routes or even changing traffic orientation in streets and avenues [TORK 2012], [Kisilevich et al.]. There is also the proposal of different clustering algorithms such as [Neill 2006], [Birant and Kut 2007] and [Rocha et al. 2010]. However, it was not found any work that proposes a broad arrangement methodology of transportation means (proposing combination of automobiles, vans and buses) directed to a group of people that share a common destination in a determined time of the day. More specifically, there is not any work that proposes a combination of transport arrangements with different capacities, for a group of students or workers that need to arrive at a specific destination (school or company) at a determined hour.

Thus, this work presents as principal contribution the proposal of a methodology of spatio-temporal clusterization, aiming to find compositions and routes of means of transportation. This proposal is compared to the actual locomotion model, in which each individual locomotion is independent of the others, using the actual public transportation.

According to [Tang et al. 2012], there are some aspects that are not adequately addressed by the works currently available in the literature. To name a few, it is possible to highlight:

- Proximity in time and space: to move together, the objects not only need to be close in space, but also need to move in similar times.
- Efficiency: the trajectories that are generated require, in fact, to reduce the locomotion times, also generating economy for the individual.
- Effectivity: many people already have a locomotion route with a robust time and cost (e.g. people who live less than 700 meters of the working place) and hardly can have their cost and time reduced. In this case, the methodology has to keep its focus on the ones who demands more time to move.

The proposed work aims to cover the questions presented by [Tang et al. 2012], and this will be detailed in the next sections. The paper is divided as follows. In section 2 it is detailed the used clusterization algorithm, besides the reasons why it was chosen as the base algorithm. Section 3 presents the modeling of grouping people problem, also as the application of an efficient solution for resolution. The section 4 contains a study-case developed, the parameters used and the results obtained. In section 5 there is a detailed analysis of the results and also a comparison with the actual scenario considered which everyone uses the public transportation. Finally, in section 6 the main conclusions of this work and the future works suggested are disposed.

2. Clusterization Algorithm - DBSCAN

The first step of the problem modeling was the choice of a clusterization algorithm. Many procedures were tested, such as K-Means and variations and the DBSCAN. The K-Means algorithm and its variations were discarded after the tests with the database. These algorithms presented big variability of the results, being dependent on the choice of the centroid, and different values yield significantly different clusters. Moreover, these algorithms presented a strong tendency to generate only spherical clusters and there were

some difficulties in estimating the K parameter (number of clusters) for the algorithm execution.

The choice of DBSCAN is justified after the satisfactory results obtained when it was applied on the studied database. Preliminary results showed the passengers were well distributed in the clusters and the number of noises was low. The details of the algorithm are presented below.

The DBSCAN algorithm uses the local density of the points to determine the clusters. For a point X, a circle of radius ϵ is defined around this point and, thus, we obtain his neighbors, as we can see in the Figure 1(a). For the existence of a cluster, it is necessary to have a minimum number of points inside this circle, i.e., it is necessary to have a minimum number of neighbors of the point X, defined by the parameter *minPts*. Thus, the algorithm iterates on the dataset, verifying dense regions and associating neighbors points to the same clusters, taking in count the condition of the minimum number of points by cluster (Figure 1(b)).

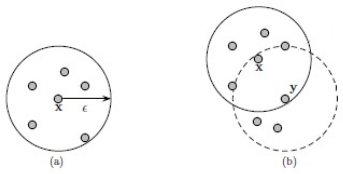


Figura 1. a) Point X and his neighborhood, by) Neighborhood expanded with the neighbor points of Y - Adaptation of [Zaki and Jr. 2013]

Since the algorithm considers the density of the points, not always the entire set of points will be associated to a cluster. There can be isolated points, in such a way that there are not a minimum number of neighbors to form a cluster. These points are considered as noises.

Thereby, the algorithm consists of four steps:

1. All the point are initially associated to a cluster -1 and are marked as not visited.
2. Region Query Method: this method makes the search to find the neighbors of the actual point, called region query. The algorithm starts iterating over the points of the database and for each not visited point, the neighbors of this points are analyzed, according to the radius ϵ. If the number of neighbors of this point is higher than the minimum of points defined by the algorithm (parameter *minPts*), this point is associated to the actual cluster (the number of the cluster starts in -1 and is always incremented if this condition is satisfied) and all his neighbors are associated to the same cluster and marked as visited. If the number of neighbors of the points is lower than the parameter *minPts* defined, this point is considered as a noise and marked as visited.
3. Expand Cluster Method: for each point that was associated with a given cluster, the algorithm searches for neighbors of this point, which are inside a given radius ϵ, and associating them with the current cluster, marking them as visited. Thus, the current cluster is expanded.
4. The steps 2 and 3 are repeated until all point are visited and, thus, the final clusters are obtained and the noises too.

The main advantages of DSBCAN are: (1) The number of clusters do not have to be defined initially, differently from K-Means algorithm; (2) The algorithm is not so sensible to the incoming order of the points; (3) It can be defined the minimum number of elements per cluster; (4) There is the concept of Noise value, eliminating points of the database that are not relevant to the analysis.

Amongst the main disadvantages, it is possible to highlight: (1) The algorithm is sensible to the choice of ϵ parameter: if a big value is chosen we can have few clusters, and, choosing a low value, we can have many noises; (2) It is not so efficient when we have regions that are not so dense.

The DBSCAN algorithm was modified and adapted to the problem. After the definition of each cluster elements, the following steps were done: (1) Calculation of the centroid of each cluster: the DBSCAN algorithm does not works with the centroid concepts. However, after the algorithm execution, the centroids were calculated as the mean of the points of each clusters, obtaining a central point with the mean latitude and longitude of each point of the cluster; (2) Optimization and allocation of the points: with the definition of the centroid concept, some points can change from one cluster to another, since they are near to a new defined centroid. Thus, for each point, it was verified its distance in relation to the centroid, guaranteeing that they was associated to the nearest centroid.

The pseudocode for those changes, can be seen in Algorithm 1 and Algorithm 2:

```
Data: Points, Latitude and Longitude
Result: Cluster, Centroids
for i ← 0 to i ← NumberOfClusters do
    for j ← 1 to j ← NumberOfClusterPoints do
        ClusterLatitude(i) += PointLatitude(j);
        ClusterLongitude(i)+=PointLongitude(j);
        numberOfElements++;
    end
    ClusterLatitude(i) = ClusterLatitude(i) / numberOfElements;
    ClusterLongitude(i) = ClusterLongitude(i) / numberOfElements;
    numberOfElements = 0;
end
```

Algorithm 1: Step 1 - Calculate the cluster centroids

```
Data: Points, Latitude and Longitude
Result: Points, Allocation
for i ← 1 to i ← NumberofPoints do
    for j ← 0 to j ← NumberOfClusters do
        if HaversineDistance(i,CentroidOfCluster(j)) ≤
        HaversineDistance(i,CentroidOfOriginalCluster) then
            Move point i, to Cluster j;
        end
    end
end
```

Algorithm 2: Step 2: Optimization and allocation of the points

3. Problem Modeling

In order to limit the problem scope and to find a solution that improves the daily locomotion time of people, the problem was divided into two parts.

In the first part of the problem, people were divided into clusters, according to the algorithm DBSCAN. For each cluster (C_i), a centroid was calculated (K_i) and was defined as departure point for people in that determined cluster. Thus, in the first part of the locomotion route, each people (X_i) walk to this central point, spending one initial walking time (Tc_i).

It was defined that the walking time can be no more than 20 minutes. The distance of each individual (X_i) to the centroid was calculated using the Haversine Function and the the locomotion time estimate until the centroid (Tc_i) was defined as the average walking time of 3 miles/hour per person. For example, considering the distance of a determined person to the centroid of its cluster of 0.5 miles, the person would walk for about 10 minutes until reaching the centroid.

In the second part of the problem, after all people walked to the centroid of its respective cluster, the route would be done from one or more vehicles. The type of the vehicle(s) was chosen according to the number of people allocated in each cluster, as below:

- Up to 5 persons - car;
- 6 to 15 persons - van;
- 16 to 40 persons - bus;
- More than 40 persons - combination of vehicles, according to the number of persons.

Thus, after the choice of the most adequate vehicle, the rest of the route would be done without stops until the final destination. The locomotion time from each cluster to the destination is given by (T_Kj). The locomotion time of each vehicle until the final destination was calculated using Google Maps data, obtaining the time directly from the route selected by Google algorithm.

Finally, the obtained time was compared to the actual time spent by using public transportation (% of saved time). For an estimate of the actual time spend for each person, it was used the Google Maps data, considering the public transportation (bus and/or subway).

The problem can be modelled as follow:

$$\min totalLocomotionTime = \sum_{i=1}^{n} T_i \tag{1}$$

$subject\ to$:
$Tc_i \leq 20$ minutes,
$T_i \leq Ta_i$

in which
$T_i = Tc_i + Tk_j$,

n = number of persons,
T_i = total locomotion time of person i,
Tc_i = walking time of person i,
Tk_j = locomotion time from the centroid j to the destination.
Ta_i = actual locomotion time of person i.

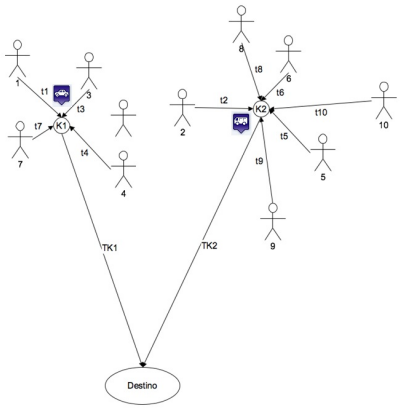


Figura 2. Proposed problem modeling, with 10 persons and 2 clusters.

Figure 2 illustrates the proposed model, considering a group of 10 persons and 2 clusters. It is possible to observe that which people were associated to each cluster, the centroid of each cluster and the type of vehicle chosen.

For the application execution it is necessary to inform the ϵ e *minPts* parameters of DBSCAN algorithm, as well as the desired arrival time at the destination for all points.

The results can be summarized into two isolated screens. The first one (Figure 3) presents the summary of the data obtained in a grid, for each cluster, containing the following information:

- For each cluster: cluster number, latitude and longitude of the centroid, number of elements, distance to the destination in meters and minutes.
- For each cluster point: point ID, latitude and longitude, distance to the centroid in meters, walking time in minutes to the centroid (also informing the departure time for each point, according to the total spending time and including a margin of 10 minutes to eventual delays), total time spent, bus time, time saved in minutes (also informing the % of time saved).

Point	[Latitude,Longitude]	Distance to the Centroid	Walking Time	Total Time	Bus Time	Saved Time
Cluster 0 -19.8845582,-43.9245864 [Qtd.:14] - Distance to Cefet Campus II: 12,573.00m - Time to Cefet: 19.77min						
1	[-19.8837,-43.9156]	930.87m	11.17min (Left home at: 6:19AM)	30.94min	70min	38.74min (% of Time Saved: 55.60%)
7	[-19.8827,-43.9299]	606.16m	7.27min (Left home at: 6:23AM)	27.04min	68min	40.59min (% of Time Saved: 60.02%)

Figura 3. Grid with the data obtained, example of cluster 0 and visualization of two points

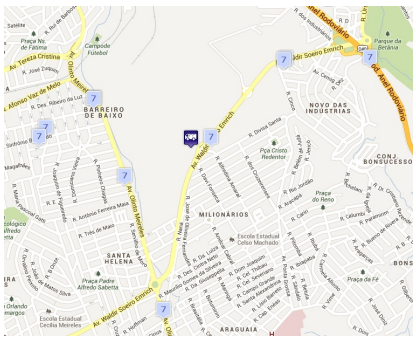


Figura 4. Visualization of the cluster obtained and its points on the Map, example of cluster 7, with 10 points (some of them very close to each other)

Figure 4 shows how the data was presented on Google Maps, divided by clusters.

According to Figure 5, the first step for the clusterization is the selection or extraction of characteristics to be studied. For our problem three characteristics were chosen: latitude, longitude and arrival time at CEFET-MG Campus II.

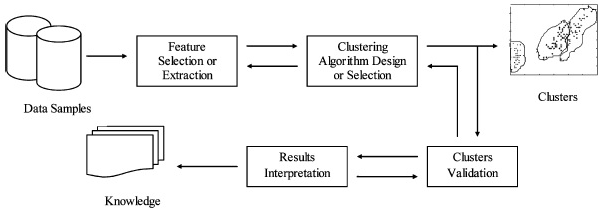


Figura 5. Clusterization Procedure - [Xu and Wunsch 2005]

For tests and validation, a database containing the localization (latitude and longitude) of 437 persons was used. In order to obtain relevant information to the problem and the validation of the final results obtained, the *Directions* API of Google Maps was used. Particularly, the API was used to establish the routes and the time required to run them, using car (or van), bus or walking, when necessary.

4. Experiment and Results

For the execution of the experiment, the following values were chosen for DBCSAN algorithm: ϵ = 1500 meters and *minPts* = 5. The destination chosen was the Centro Federal de Educação Tecnológica de Minas Gerais Campus II - CEFET-MG in Belo Horizonte (latitude = -19.9377, longitude = -43.9997) and the arrival time at destination was defined as 7 AM.

The value of ϵ was chosen according to the problem modeling. Since $Tc_i \leq 20$ minutes, using $\epsilon = 1500$ meters, the locomotion time of the neighbors to the central point is 18 minutes at most, according to the walking time defined. The value of *minPts* was chosen based on the maximum number of passengers that can be transported in a car (the mean of transportation with the minimum of passengers) and to optimize the transportation, aiming always to transport the maximum number of people per vehicle.

The experiment generated 33 clusters, transporting an total of 394 persons, which represent about 90.2 % of the total. The other 43 persons were treated as noises, since they were in low density regions, not covered by the algorithm.

The points were distributed according to the density of each region, covering all the means of transportation available. There were generated clusters varying from 5 up to 70 passengers. According to the ϵ parameter defined, the biggest walking locomotion time was 23.04 minutes, from point 184 to the center of cluster 5. This value is acceptable, since it is only 3 minutes higher than the previous value defined (15% variation).

In Figure 6 it is shown the clusters generated and how they were distributed on the map. The figure shows the distribution of the clusters generated in the city of Belo Horizonte and the mean of transportation defined for each cluster.

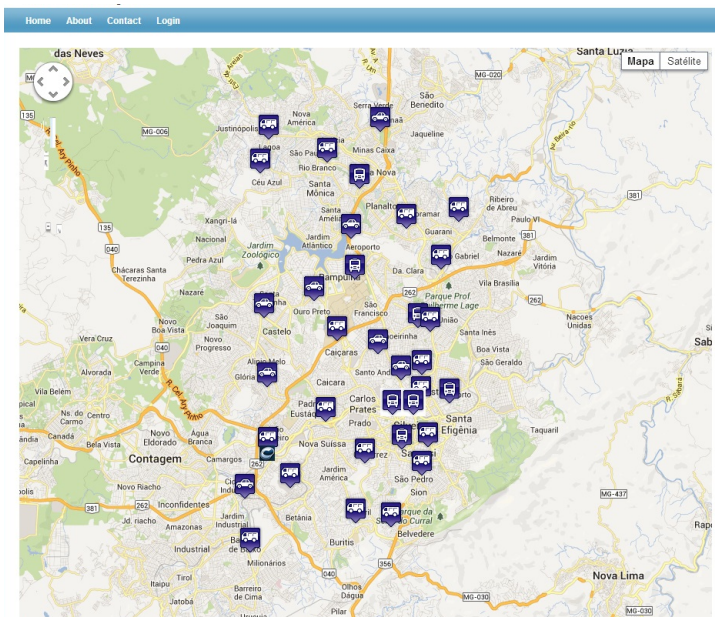


Figura 6. Clusters obtained in the application

Table 1 summarizes the clusters found, showing the number of elements obtained on each one and the average time saved by each cluster. It is important to highlight that, to compute the average saved time, it was considered the walking time necessary to each person leave its home and arrives in the stop point. In the first case, using the conventional public transportation, people walk from their home to the bus stop; in the second case, using the proposed approach, people walk from their home to the bus/van/car stop point (centroid of the cluster). In both cases, the Google Maps API was used to determine the walking time of each person.

5. Analysis of Results

Through the analysis of results, we can verify that the cluster which biggest saved time is was the cluster 32, with an average of more than 1 hour and 10 minutes saved per person. The worst result, and the only cluster where there was not saved time, was the 15, with an average delay of 2 minutes and 40 seconds per person.

Tabela 1. Results obtained with DBSCAN algorithm, ϵ = 1500 meters and *minPts* = 5

Cluster	Number of Elements	Average Time Saved
0	14	45.57
1	25	49.58
2	9	33.58
3	8	39.48
4	11	22.35
5	70	16.43
6	38	23.56
7	10	9.53
8	7	54.73
9	7	56.11
10	9	43.31
11	20	37.77
12	6	19.51
13	23	64.41
14	16	23.67
15	7	-2,69
16	5	2.62
17	5	56.21
18	10	24.80
19	6	25.29
20	5	37.96
21	6	54.48
22	5	53.59
23	5	28.89
24	5	25.84
25	6	26.65
26	11	21.00
27	5	53.88
28	10	67.71
29	5	55.85
30	7	33.29
31	10	67.51
32	8	70.30
Total	394	34.51
Noises	43	

Analyzing Table 2, we can verify how the points were distributed on the cluster 32. In Figure 7 it is highlighted the cluster 32 with its points (in the top) and the localization of CEFET-MG (in the bottom), being possible to verify the distance between them. Table 3 shows the data from cluster 15. The points of cluster 15 are shown on the map in Figure 8.

Tabela 2. Points distributed on the cluster 32

Point	Centroid Distance	Walking Time	Total Time	Bus	Saved Time (% saved)
221	662.68m	7.95min	38.40min	108min	69.97min (64.56%)
230	662.64m	7.95min	38.40min	108min	69.97min (64.56%)
329	779.54m	9.35min	39.80min	99min	58.80min (59.63%)
330	590.08m	7.08min	37.53min	105min	67.52min (64.27%)
331	376.96m	4.52min	34.97min	119min	84.08min (70.62%)
332	821.30m	9.86min	40.31min	106min	65.42min (61.88%)
338	789.05m	9.47min	39.92min	114min	74.20min (65.02%)
344	532.90m	6.39min	36.84min	109min	72.46min (66.29%)

Verifying the location of the clusters on the map, we can see that the result obtained on the cluster 15 is justifiable, since all points of these clusters are really close to CEFET-MG. For this dataset, the proposed solution would not be adequate, since for those people is better to walk directly to CEFET-MG than walking to the centroid and

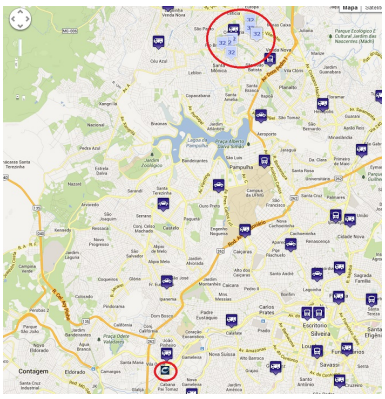


Figura 7. Visualization of cluster 32, its points and the distance in relation to CEET-MG

Tabela 3. Points distributed on the cluster 15

Point	Centroid Distance	Walking Time	Total Time	Bus	Saved Time (% Saved)
30	368.90m	4.43min	13.81min	10min	-4.13min (-42.67%)
32	489.54m	5.87min	15.26min	32min	16.51min (51.97%)
65	364.30m	4.37min	13.75min	10min	-3.82min (-38.52%)
80	505.91m	6.07min	15.45min	3min	-12.52min (-427.45%)
81	507.18m	6.09min	15.47min	1min	-14.69min (-1883.27%)
154	1,125.60m	13.51min	22.89min	33min	9.74min (29.85%)
177	372.29m	4.47min	13.85min	4min	-9.93min (-253.34%)

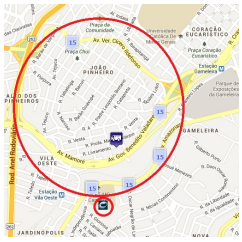


Figura 8. Visualization of cluster 15, its points and the distance in relation to CEET-MG

then get the public transportation.

On the other hand, for the cluster 32, localized far away from CEFET-MG, we can check the efficacy of the proposed solution. The average time saved for this cluster is really high and, in the worst case, the saved time is 58.80 minutes, which represents an economy of 59.63% of the previous time spent by this person using public transportation.

In the Figure 9, we can see how the saved timed is strongly related to the distance of the points and CEFET-MG. For the points highlighted in green, we obtained the highest saved times: 70.30 minutes for the cluster 32, 67.71 minutes for the 28, 67.51 minutes for the 31, 64.41 minutes for the 13 and 56.21 minutes for the 17. For the points in the red region, we obtained the worst saved times: -2.69 minutes for the cluster 15, 2.62 minutes for the 16 and 9.53 minutes for the 7.

In this way, we can conclude that the factor which has the most influence on the saved time for a point is its distance to the destination. For points relatively far from the destination, the efficiency of the algorithm is really high, with an average save time

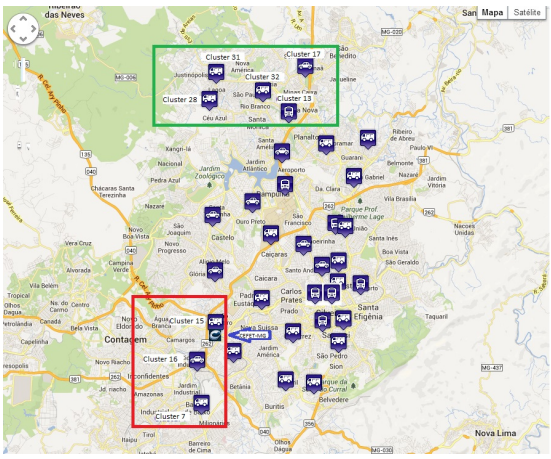


Figura 9. Visualization of the clusters with higher saved time (green) and smaller saved time (red).

closer to 65%. On the other hand, for points closer to the destination, the efficiency of the algorithm is lower, and can be negative in points really closer to the destination. For intermediate points, the average saved time is really satisfactory, with an average of 34.51 minutes saved per person, which represent an economy rate of 45.80% in the average.

Finally, some points are marked as noises as can be seen in Figure 10. We can verify that those points are more isolated, being unable to associate them to a cluster. In some case, despite of having close points, they do not meet the requirement of the parameter *minPts* to form a new cluster.

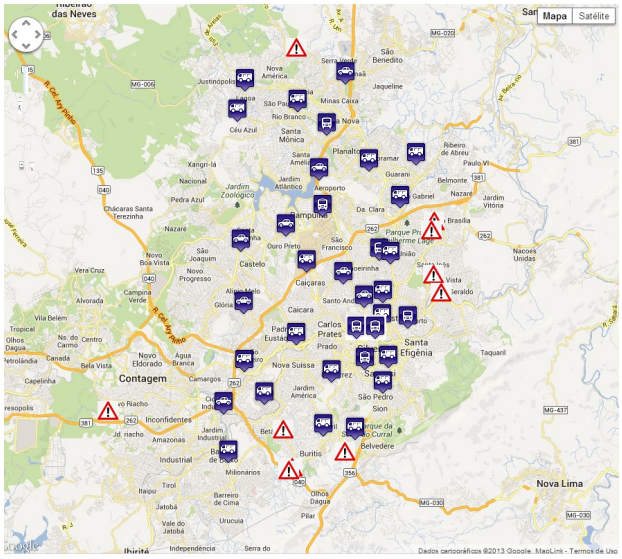


Figura 10. Some points marked as noises

6. Conclusion and Future Works

According to the obtained results, we can conclude that the problem modelling and the application developed have reached the proposed objectives on the work. The result obtained has a highly relevance, since it not only can helps the reduction of people daily

locomotion time, but also contributes to the reduction of traffic in the cities, encouraging means of public transportation or carpooling directed to the needs of their users. The application can be used in transportation of large masses of people to a specific location. Certainly, an average economy of 45.80% of locomotion time is something really relevant, which shows that simple solutions well-modeled can help to reduce one of the metropolises big problems, that is the traffic and loss of time in locomotion. Thus, the main contributions of this paper are: (1) An efficient application to reduce the daily locomotion time of a large number of persons until a common destination; (2) To incentive collective transportation (bus, vanpooling and carpooling), reducing the number of vehicles on the streets and contributing for a better traffic; (3) Adaptation of a clustering algorithm and modeling of an application to solve traffic problems.

As a continuity of this work, the following issues are proposed: (1) Algorithm improvements to work with Google distance function, obtaining traffic data during the displacement; (2) Pre-processing of the data using the UTM (*Universal Transverse Mercator*) coordinates system in order to simplify the distance calculation; (3) Insertion of time factor on the algorithm, working with people locomotion in different hours.

Referências

Birant, D. and Kut, A. (2007). St-dbscan: An algorithm for clustering spatial temporal data. *Data and Knowledge Engineering*, 60(1):208 – 221.

Kisilevich, S., Mansmann, F., Nanni, M., and Rinzivillo, S. Spatio-temporal clustering: a survey.

Neill, D. B. (2006). Detection of spatial and spatio-temporal cluster. *School of Computer Science, Carnegie Mellon University*.

Newton, R. M. and Thomas, W. H. (1969). Design of school bus routes by computer. *Socio-Economic Planning Sciences*, 3(1):75 – 85.

Park, J. and Kim, B.-I. (2010). The school bus routing problem: A review. *European Journal of Operational Research*, 202(2):311–319. cited By (since 1996)38.

Rocha, J., Oliveira, G., Alvares, L., Bogorny, V., and Times, V. (2010). Db-smot: A direction-based spatio-temporal clustering method. In *Intelligent Systems (IS), 2010 5th IEEE International Conference*, pages 114–119.

Tang, L.-A., Zheng, Y., Yuan, J., Han, J., Leung, A., Hung, C.-C., and Peng, W.-C. (2012). On discovery of traveling companions from streaming trajectories. pages 186–197.

Tang, L.-A., Zheng, Y., Yuan, J., Han, J., Leung, A., Peng, W.-C., and Porta, T.-L. (2013). A framework of traveling companion discovery on trajectory data streams. *ACM Transactions on Intelligent Systems and Technology*, 5(1).

TORK, H. F. (2012). Spatio-temporal clustering methods classification. *DSIE - Doctoral Symposium in Informatics Engineering*.

Xu, R. and Wunsch, D. C. (2005). Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, 16(3):645–678.

Zaki, M. J. and Jr., W. M. (2013). *Data Mining and Analysis: Fundamental Concepts and Algorithms*.

Annotating Trajectories by Fusing them with Social Media Users' Posts

Ricardo Gil Belther Nabo[1], Renato Fileto[1], Mirco Nanni[2], Chiara Renso[2]

[1]Dept. of Statistics and Informatics (INE), Santa Catarina Federal University (UFSC), University Campus of Trindade, 88040-900, Florianópolis-SC, BRAZIL

[2]Institute of Information Science and Technologies "Alessandro Faedo" (ISTI), National Research Council (CNR), Via G. Moruzzi, 1, S. Cataldo, 56124, Pisa, ITALY

Abstract. *The widespread use of mobile devices allows gathering large amounts of moving objects' trajectories. However, just trajectories are often not enough to enable movements understanding. On the other hand, users' posts in social media can be regarded as sparse and freely annotated movement traces, which can also be collected via mobile devices. This paper proposes a method for automatically fusing trajectories with social media users' posts based on their spatiotemporal compatibility. The results are trajectories annotated with posts contents, that may help to explain movement goals, and relations with places and events, among other information. The proposed method has been implemented and evaluated in experiments with real GPS trajectories and tweets.*

1. Introduction

The popularization of mobile devices equipped with positioning technologies (e.g. GPS navigators, smartphones, tablets) has increased considerable in the recent years. This growth has led to the gathering of large volumes of raw trajectories, i.e., time ordered sequences of spatiotemporal positions of moving objects holding mobile devices. The trajectories collected by using modern devices can have quite accurate spatiotemporal coordinates (e.g., 3 to 30 meters), which are collected in such a rate (e.g., at each second) that allows capturing many movement details. However, such purely spatiotemporal data lacks information (e.g., in the form of textual data) to help understand the movements, such as related places of interest, events, and goals.

Several works have been proposed for trajectories data processing and mining [Spaccapietra et al. 2008, Alvares et al. 2007, Parent et al. 2013, Pelekis and Theodoridis 2014], but spatiotemporal coordinates are not enough to explain movements [Yan et al. 2013, Fileto et al. 2013], making trajectories annotation crucial to realize their information analysis potential. Thus, many solutions have been proposed for trajectories annotation. Nevertheless, these methods have limitations on the characteristics of the annotations produced and/or rely on human labor. The former problem limits the use of the generated annotations. The latter makes the methods unsuitable for daily use with large quantities of trajectories, because annotating is a laborious task, that can easily become tedious for people. On the other hand, the sparse spatiotemporal data available in social media (e.g. Twiter, Facebook), have textual information (e.g., hashtags, comments) that can help describe and analyze trajectories. These data can be regarded as sparse and freely annotated movement traces [Azmandian et al. 2012], and

are also frequently collected via mobile devices. Thus, they can be used to annotate trajectories and analyse movements.

This paper proposes a method to annotate raw trajectories by fusing them with social media users data (e.g. tweets, posts in Facebook). It uses as the matching criteria the proximity of each trajectory (or trajectory segment) with sequences of social media users' posts. Our strategy is analogous to some of those that are common practice in many engineering areas to fuse data originated from different kinds of devices [Castanedo et al. 2010]. The resulting trajectories annotated with textual contents of social media posts (hashtags, comments, etc.) can feed semantic enrichment, analysis, and mining processes for extracting useful information of large quantities of spatiotemporal data. In fact, some collections of sparse, geo-referenced, and timestamped textual contents of social media users' posts are already being proved useful to explain goals, places, and events related to movements of people [Fileto et al. 2013, May and Fileto 2014]. The method proposed in this paper aims to ally the virtues of vast collections of trajectories (usually accurate, detailed) and social media posts (having rich textual contents). It has been implemented in a prototype, and evaluated in experiments with real GPS trajectories and tweets, both with geographic coordinates inside the city of Fortaleza Brazil.

The remaining of this paper is organized as follows. Section 2 discusses related works. Section 3 defines some key concepts for understanding the proposal. Section 4 presents the proposed method. Section 5 describes some experiments and their results. Finally, Section 6 concludes the paper, and provides a glimpse of our future work.

2. Related Works

One of the most accepted conceptual models for structuring trajectories is proposed by [Spaccapietra et al. 2008]. According to this model trajectories can be segmented in stops and moves, which are specializations of episodes (Definition 3). It introduces the possibility of interpreting a subsequence of spatiotemporal points (stop or move) as an aggregation with distinguishable characteristics. It abstracts irrelevant details, and allows annotations to be associated with stops and moves instead of trajectory points.

The method proposed by [Alvares et al. 2007] aims to semantically enrich trajectories by mining stops in places of interest (POI) of a given collection, and moves between these stops. Their method allows the efficient calculation of stops and moves, to build the conceptual representation trajectories proposed in [Spaccapietra et al. 2008]. This method, originally called SMoT (Stops and Moves of Trajectories), can also be called IB-SMoT (Intersection-Based SMoT). CB-SMoT (Cluster-Based SMoT) [Xiu-li and Wei-xiang 2009] is another method to mine stops and moves. It aggregates spatiotemporal points of raw trajectories in subsequences that present similar characteristics (e.g., around the same speed). CB-SMoT can identify stops by clustering adjacent positions in which the moving object is stationary or moves slowly, regardless of where they occur. Notwithstanding, IB-SMoT and CB-SMoT produce limited annotations. Both label segments as stops or moves, and IB-SMoT associates each stop with the respective POI of the given collection.

The annotation platform proposed in [Yan et al. 2013] progressively transforms the raw trajectories into semantic trajectories. The trajectory segments are annotated with concepts such as as home and work, or POIs. These annotations are based on prede-

termined hot spots, and produced by trajectory mining algorithms. They derive from behavior found in trajectories and/or external data.

The DayTag annotation system [Rinzivillo et al. 2013] helps an individual to reconstruct her/his travel diary from the GPS trajectories collected by using a smartphone. The user uploads his trajectories and interacts with the system to visualize and annotate trajectories. It generates diaries a posteriori, instead of annotating trajectories during real time on mobile devices, as done in works like [Doulamis et al. 2012, Broll et al. 2012]. These tools sometimes they infer basic information, such as the kind of transportation means (motorized vehicle) or nearby places, by using spatiotemporal analysis or checking the places that are close to the user's location in available databases. However, these semi-automatic annotation tools still demand a lot of user effort to confirm what is inferred, and mainly to provide additional information for annotations (e.g., places and events of interest, goals). Our work, on the other hand, proposes a totally automatic method that can be applied to huge data volumes, without demanding additional user effort.

The movement mining algorithms presented in [Azmandian et al. 2012] use as inputs another source of movement data: sequences of social posts. It examines the movement patterns of Twitter users and cluster moving objects according to their spatiotemporal these patterns. The results of this work show that it is possible to infer part of the underlying transportation network from Tweets alone, and uncover interesting differences between the behaviors exhibited by users across cities. [Gabrielli et al. 2013] exploits mobility data mining techniques along with social network analysis methods to aggregate similar trajectories, and point out hot spots of activities, and flows of people that vary over time, according to the number of tweets sent from each place. They apply and validate the proposed trajectory mining approaches to a large set of trajectories built from geo-positioned tweets gathered in Barcelona during the Mobile World Congress 2012.

Other works dealing with social media posts to peoples activities and behavior as they move in the geographic space include [Kisilevich et al. 2010, Cheng et al. 2011, Yin et al. 2011, Zigkolis et al. 2011, Wakamiya et al. 2012]. However, none of these proposals use the textual contents of social media posts to annotate trajectories or to help explain movements.

3. Basic Definitions

This section first presents definitions related to trajectories, and social media users' trails (i.e., sequences of social media user's posts). Then, it describes the problem of fusing them to produce trajectories annotated with the textual contents of the posts. These subjects are fundamental to understand the rest of the paper.

3.1. Trajectories

Raw trajectories are temporally ordered sequences of spatiotemporal positions occupied by a moving object. In this work, we consider trajectories of small objects (e.g., people, vehicles), whose positions are represented by spatiotemporal points.

Definition 1. (**Spatiotemporal Point**). Position represented by the quadruple: $p(p\_id, x, y, t)$, where:

- p_id is a point identifier;

- (x, y) is a pair of geographic coordinates; and
- t is a time instant.

A mobile device that collects locations samples in the form of spatiotemporal points within a certain time interval generates a raw trajectory.

Definition 2. (**Raw Trajectory**). Temporally ordered sequence of spatiotemporal positions visited by a moving object, represented by the triple: $RawTraj(mo\_id, t\_id, p\_seq)$, where:

- mo_id is the mobile object identifier;
- t_id is the trajectory identifier; and
- p_seq is a temporally ordered sequence of spatiotemporal points $(p_1, \ldots, p_n)$, with each p_i $(1 \leq i \leq n)$ of the form stated in Definition 1.

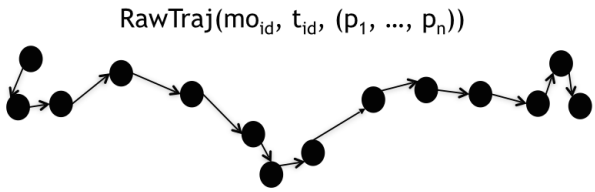


Figure 1. Raw Trajectory

A raw trajectory can be segmented in subsequences of spatiotemporal points satisfying certain conditions. These subsequences are called episodes, as defined in the following.

Definition 3. (**Episode**). Maximal subsequence of spatiotemporal points of a raw trajectory that satisfies a given predicate. An episode is represented by the quadruple: $episode(t\_id, e\_id, e\_type, p\_subseq)$, where:

- t_id is the trajectory identifier;
- e_id is the episode identifier;
- e_type is the episode type (e.g. $Stop$,$Move$); and
- p_subseq is a maximal subsequence of spatiotemporal points $(p_i, \ldots, p_j)$ from a raw trajectory $RawTraj(mo\_id, t\_id, (p_1, \ldots, p_n))$ that satisfies the predicate $(p_i, \ldots, p_j) \implies \{true, false\}$ $(1 \leq i \leq j \leq n)$.

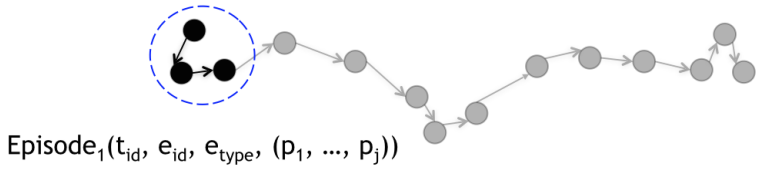


Figure 2. Episode

Temporally ordered episodes of a trajectory, constitute another representation of the movement, called a structured trajectory, as defined in the following.

Definition 4. (**Structured Trajectory**). Temporally ordered sequence of non nested episodes. Each element of the sequence is represented by a pair: **StrTraj**$(stid, Ei)$, where:

- $STid$ is the structured trajectory identifier; and
- Ei is an episode.

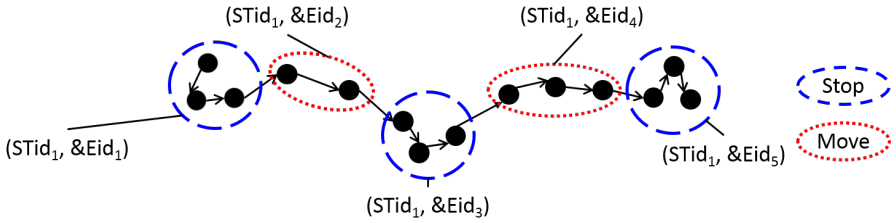


Figure 3. Structured Trajectory

3.2. Movement Data collected on Social Media

A social media footprint is the record of an interaction between an user and a social media (eg, Twitter, Facebook, Foursquare). When the user posts an associated information (eg, space-time position, photo) is recorded in the respective media (eg, Twitter, Facebook) and accessible via specific each media API. A temporally ordered sequence of footprints is a trail.

Definition 5. (**Social Media Footprint**). Social media system record of an iteration performed by a user, represented by the quintuple: **SMF**$(MOid, SMFid, Smid, P, c)$, where:

- $MOid$ is the mobile object identifier;
- $SMFid$ is the footprint identifier;
- $SMid$ is the social media identifier (e.g., Twitter, Facebook);
- P is a reference to a spatiotemporal point (Definition 1);
- c are the contents of the footprints (e.g., tags, pictures, texts).

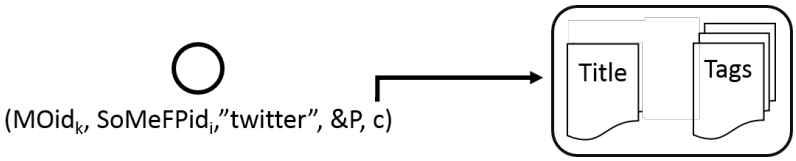


Figure 4. Social Media Footprint

Definition 6. (**Social Media Trail**). Temporally ordered social media footprints sequence, generated by the same user. Each element of this sequence is represented by the pair: **SMT**$(SMTid, SMF)$, where:

- $SMTid$ is the social media trail identifier; and
- SMF is a reference to a social media footprint (Definition 5).

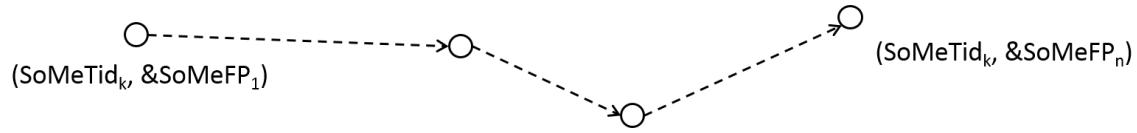


Figure 5. Social Media Trail

Both raw trajectories and social media trails refer to sequences of spatiotemporal positions. However, trajectories usually have better spatiotemporal accuracy than trails. Raw trajectory points are usually sampled at short and fixed intervals (e.g., every second, every 10 meters). On the other hand, social media posts are asynchronous (the user decides when to post) and usually sparse, but they have associated textual contents, that may serve as annotations to help understand movements.

The problem considered in this paper is the fusion of (portions of) trajectories with (portions of) trails, based on spatiotemporal proximity, to produce trajectories annotated with trails contents. Its inputs are a set of raw trajectories and a set of social media trails. Its outputs are pairs of the form $\langle traj, trail \rangle$, where $traj$ is a moving object's trajectory or a continuous subsequence of its points, and $trail$ is a sequence of social media user's footprints (posts). Each returned pair $\langle traj, trail \rangle$ must have trajectory points and trail footprints that are close in space and time, as illustrated in the upper portion of Figure 7.

4. Proposed Method for Fusing Trajectories with Trails

The proposed method efficiently determines the best matching pairs $\langle traj, trail \rangle$ from a large dataset of trajectories and trails, by using a spatiotemporal distance function that allows ranking the matchings, among those that satisfy at least the minimum matching criteria. Figure 6 presents an overview of the proposed method. It is a process with three phases: trajectories preprocessing, trajectories compression, and fusion of trajectories with trails. The preprocessing phase can clean and structure raw trajectories in a sequence of episodes, for example. The compression stage compresses the structured trajectories in a representation that can be analyzed more efficiently than the mere aggregation of the trajectory points in episodes. The fusing phase calculates the global matching coefficient for pairs $\langle traj, trail \rangle$ that may be related and selects the best ones. This method is flexible in the sense that it allows different algorithms for performing specific tasks in each phase, according to the dataset and application peculiarities. The main focus of this paper is the fusion phase, that is divided in four steps, described in the following subsections.

4.1. Select Candidate Pairs

Comparing every pair $\langle traj, trail \rangle$ is not viable for large datasets, due to the amount of time needed for doing so. Thus, we consider a temporal window $[t_i, t_f]$ around each trajectory and trails, where t_i is the initial instant of the trajectory trail less a threshold, and t_f is the final instant of the trajectory or trail plus the same threshold. Using these temporal windows it is possible to efficiently select only the pairs $\langle traj, trail \rangle$ that temporally overlap. It is also possible to use spatio-temporal windows and joins based on their intersections processed with efficient spatiotemporal data access methods to determine the candidate matching pairs. The matching based on enclosing windows is expected to generate a relatively small set of pairs $\langle traj, trail \rangle$, compared to the Cartesian product of the trajectories and trails datasets. These pairs are then evaluated in mored detail to calculate

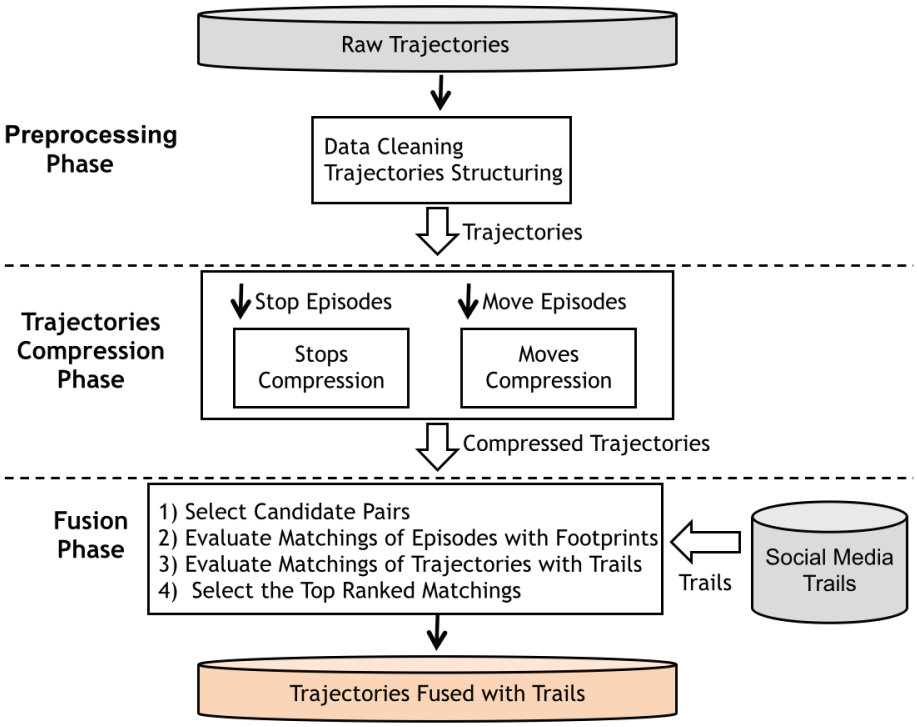


Figure 6. General process of the proposed method

first the local matching coefficients between trajectory episodes and trail footprints, and then the global matching coefficients of the respective $\langle traj, trail \rangle$ pairs.

4.2. Evaluate Matchings of Episodes with Footprints

Consider a candidate pair $\langle traj, trail \rangle$ such that $traj$ is a time ordered sequence of episodes $e_1, \ldots, e_n$ of a moving object's structured trajectory, and $trail$ is a time ordered sequence of footprints (posts) $f_1, \ldots, f_m$ of a social media user's trail ($m, n \geq 1$). The **Temporal Matching Coefficient (TMC)**, given by Equation 1, measures the temporal compatibility between episode e_i and footprint f_j ($1 \leq i \leq n$, $1 \leq j \leq m$).

$$TMC(e_i, f_j) = \begin{cases} 0 & \text{if } (f_j.t \in e_i.t) \\ |\Delta t(e_i.t, f_j.t)| & \text{if } (|\Delta t(e_i.t, f_j.t)|) \leq \tau_t \\ \infty & \text{otherwise.} \end{cases} \quad (1)$$

TMC is 0 if the time stamp $f_j.t$ of the footprint f_j is inside the time span of episode e_i. Otherwise, TMC is the time difference between the trajectory and the trail, if this difference is less or equal an predetermined time threshold τ_t. If this difference is greater than τ_t then TMC is set to infinity. This coefficient guaranties that the footprint is temporally close to the trajectory episode that it may be associated to. It is crucial in the proposed method, because its goal is to associate episodes with social media posts that occur around the same time, and the time stamps are usually reliable in both data sources.

The **Spatial Matching Coefficient (SMC)** of Equation 2 employs a distance metric of the Minkowsky family (L_p) to measure the spatial compatibility between a trajectory episode e_i and a trail footprint f_j.

$$SMC_k(e_i, f_j) = \sqrt[k]{|(e_i.x - f_j.x)|^k + |(e_i.y - f_j.y)|^k} \quad (2)$$

Variations of Equations 1 and 2 could use, for example, the log of δ_t and Lp, respectively, instead of the bare distances. It can help to adjust the measurement scales in a more suitable manner to capture the temporal and/or spatial compatibilities for certain datasets and application domains.

Finally, the **Local Matching Coefficient (LMC)** between trajectory episode e_i and trail footprint f_j is calculated as stated by Equation 3, which simply sums the values $TMC(e_i, f_j)$ with $SMC(e_i, f_j)$, calculated by using Equations 1 and 2, respectively.

$$LMC(e_i, f_j) = TMC(e_i, f_j) + SMC(e_i, f_j) \quad (3)$$

4.3. Evaluate Matchings of Trajectories with Trails

The Global Matching Coefficient (GMC) of a pair $\langle traj, trail \rangle$ is calculated by using the LMC between each spatiotemporally close episode and footprint of $traj$ and $trail$, respectively. In this work, we use Dynamic Time Warping (DTW) [Rakthanmanon et al. 2013] for doing this task. DTW is an efficient algorithm to calculate the proximity between two temporal sequences, by computing optimal matchings between their component points. The sequences are warped non-linearly in the time dimension to determine a measure of their similarity, independent of certain non-linear variations in the time dimension. Although DTW measures a distance-like quantity between two given sequences, it does not guarantee the triangular inequality property.

In this work $GMC(traj, trail)$ is the DTW proximity between $traj$ and $trail$, i.e., the optimal sum of $\sum_{e_i \in traj} LMC(e_i, f_j)$ between episodes of a trajectory and trail footprints. DTW has two binding possibilities to an episode and a footprint, regarding annotation purposes. These cases are: (i) bind an episode with 1 or more footprints of a trail (B1+); or (ii) bind more than one episode of a trajectory with the same footprint (B+1). In case B1+ we annotate the episode with all footprints binded to it, and in case B+1 we annotate the episode with the closest footprint. These binding cases are illustrated in the figure 7.

4.4. Select the Top Ranked Matchings

After computing the GMC between $\langle traj, trail \rangle$ pairs, these pairs are regarded as edges of a bipartite graph $BG(V, E)$ where $V = TrailsSet \cup TrajectoriesSet$, and the weight of each edge $\langle traj, trail \rangle$ is the value $GMC(traj, trail)$. Then, a greedy algorithm that orders the edges in descending order of their weights, and takes the edge (pair $\langle traj, trail \rangle$) with the lowest weight (value $GMC(traj, trail)$) to annotate episodes of the trajectory $traj$ with the textual information associated to the footprints of $trail$.

5. Experiments

We have implemented the method proposed in this work as a prototype. The implementation of this prototype was done in Java version 1.7.0. The database management system

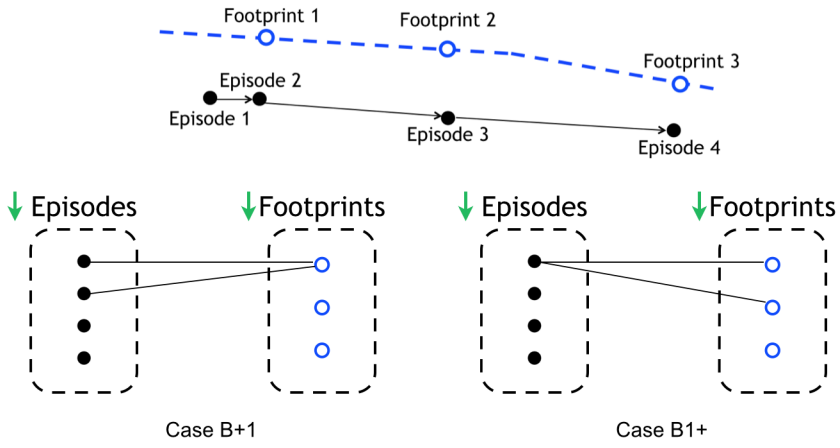


Figure 7. Illustration of the binding cases for a pair $\langle traj, trail \rangle$.

Postgres[1] version 9.1.3 and its spatial extension PostGIS[2] version 2.1 were used to hold trajectory and social media data. They provide support to efficiently access and process this data, with operators such as `CONTAINS`, `OVERLAPS`, and proximity joins that use these operators on geometric representations of spatiotemporal data indexed with GIST[3].

The proposed method needs two datasets as its inputs: a trajectories dataset and a trails dataset, that must be spatially and temporally compatible (i.e., contain data of the same geographic area, preferably collected during the same time period). However, finding compatible datasets has not been a trivial task. Currently, we only have access to trajectory databases that were collected a few years ago, while social media APIs only allow collecting data of recent posts, or even those occurring at real time. Thus, we are doing efforts in parallel to collect trajectories and posts that are compatible in space and time. In fact, we are pursuing work with volunteers to collect some subsets of trajectories connected to specific sequences of social media users' posts, to serve as ground truth to evaluate the quality of our method. Meanwhile, we have done preliminary experiments to evaluate our method with partially compatible datasets, such as collections of trajectories and social media data of the same geographic area, but not the same year. Then, in the experiments done so far, we consider datasets with temporal compatibilities such as the same days of the year, months or seasons, but in different years.

The Dataset of Raw Trajectories (DRT) was collected by using GPS on taxis moving in the metropolitan region of Fortaleza, during the period between July 3 2012 and October 20 2012. For this experiment we selected 10 taxis drivers, and segmented the data in such a way that each trajectory corresponds to a taxi ride for a passenger, generating a total of 8,253 trajectories. The Dataset of Social Media Trails (DSMT) contains the posts of 11,974 Twitter users, who sent tweets from the metropolitan region of Fortaleza between July 3 2014 and October 15 2014, making a total of 339,713 footprints.

[1]http://www.postgresql.org
[2]http://postgis.net
[3]http://gist.cs.berkeley.edu

5.1. Data Processing

In the Pre-processing Phase we deleted from the DRT all the trajectories that had less than 10 points, and assured that they were segmented by the taxis ride. These procedures reduced the number of trajectories of the DRT to 6,429. Then, we applied the algorithm CB-SMoT [Bogorny et al. 2011] to generate the Database of Structured Trajectories (DST), containing 13,167 stops in 4,962 trajectories. All the stops generated have their duration equal or bigger than 15 seconds, average movement speed of 0.5 km/h, and maximum instantaneous speed between two points of 1 km/h. The other 1,467 trajectories did not meet these requisites to generate valid stops.

In the Trajectory Compression Phase, we computed the centroids of the stops produced in the previous phase for each structured trajectory in DST. In the experiments done so far we did not use the moves of these trajectories.

Finally, in the Fusion Phase we set the temporal threshold for the temporal window to 5 minutes. It is important to denote that our trajectory dataset do not match the trail dataset exactly in time, we disregard the the year of the trajectories data. In addition, as these datasets do not have a matching segmentations, we allow matching to be done between subsequences of trails and subsequences of trajectories. Therefore, the pairs $\langle traj, trail \rangle$ do not have to match as a whole.

The average number of candidate trails to match each trajectory in the DST were 2 trails, after applying the temporal window. Thus, trails that are out of the temporal window of each trajectory are not taken into account in the computation of the GMC. Consequently, the fusing algorithm can run more efficiently by only considering time compatible trails to match with each trajectory.

We used the distance metrics L1 and L2 to calculate the spatial compatibility coefficient (SMC_k), i.e, we made experiments with $k = 1$ and $k = 2$ in Equation 2. We built as an output of the Global Matching Coefficient (GMC) computation an Associative Database (ADB) that is composed by a structured trajectory from DST, a trail id from DSMT, a GMC_1 value considering SMC_1, and a GMC_2 value considering SMC_2.

5.2. Results

We verified that 87% of the greedy algorithm choices for binding pairs $\langle traj, trail \rangle$ were the same using either SMC_1 or SMC_2. The mean execution time for the method was 8 minutes, obtained in set of 20 executions with the same conditions. We computed SMC_1 and SMC_2 in each one of these executions.

The proposed method was able to annotate around 32% of the structured trajectories that had at least one stop, using either GMC_1 or GMC_2. The execution of the method using GMC_1 and GMC_2 generated 1,621 annotated trajectories with at least one episode annotated with the textual contents of a social media post.

6. Conclusion and Future Works

This paper proposes a method to fuse moving objects' trajectories with social media users' trails (sequences of posts). The proposed process adds the textual information of posts contents to structured trajectories to produce annotated trajectories. This fusion is performed in three phases: preprocessing, trajectory structuring, and data fusion. The last

phase relies on the spatiotemporal proximity of individual posts of a trail with trajectory episodes. The main contributions of this proposal are: (i) for the best of our knowledge, it the first one to annotate trajectories via fusion with social media posts; (ii) the proposed method is totally automatic; and (iii) it is convenient and efficient enough to be used with vast amounts of trajectories and social media data.

Future works include: (i) conduct further information fusing experiments with other trajectories and social media data collections; (ii) extend the current version of the method to also annotate moves; (iii) optimize the proposed method to run faster; (iv) evaluate the quality of the annotated trajectories generated by different versions of the proposed method, and schemes for tunning its parameters; and (v) employ the resulting textually annotated trajectories to feed a variety of semantic enrichment, analysis, and mining methods.

Acknowledgments

This work was supported by the European Union IRSES-SEEK (grant 295179) and FP7-DATASIM (grant 270833) projects, CNPq (grant 478634/2011-0), CAPES, and FEESC.

References

Alvares, L. O., Bogorny, V., Kuijpers, B., de Macedo, J. A. F., Moelans, B., and Vaisman, A. (2007). A model for enriching trajectories with semantic geographical information. In *the 15th Intl. Symp. on Advances in Geographic Information Systems*, pages 22:1–22:8, New York, NY, USA. ACM.

Azmandian, M., Singh, K., Gelsey, B., Chang, Y.-H., and Maheswaran, R. T. (2012). Following human mobility using tweets. In Cao, L., Zeng, Y., Symeonidis, A. L., Gorodetsky, V., Yu, P. S., and Singh, M. P., editors, *ADMI*, volume 7607 of *LNCS*, pages 139–149. Springer.

Bogorny, V., Avancini, H., de Paula, B. C., Kuplich, C. R., and Alvares, L. O. (2011). Weka-STPM: a software architecture and prototype for semantic trajectory data mining and visualization. *T. GIS*, 15(2).

Broll, G., Cao, H., Ebben, P., Holleis, P., Jacobs, K., Koolwaaij, J., Luther, M., and Souville, B. (2012). Tripzoom: An app to improve your mobility behavior. In *11th Intl. Conf. on Mobile and Ubiquitous Multimedia*, MUM '12, pages 57:1–57:4, New York, NY, USA. ACM.

Castanedo, F., García, J., Patricio, M. A., and Molina, J. M. (2010). Data fusion to improve trajectory tracking in a cooperative surveillance multi-agent architecture. *Information Fusion*, 11(3):243 – 255. Agent-Based Information Fusion.

Cheng, Z., Caverlee, J., Lee, K., and Sui, D. Z. (2011). Exploring millions of footprints in location sharing services. In Adamic, L. A., Baeza-Yates, R. A., and Counts, S., editors, *ICWSM*. The AAAI Press.

Doulamis, A., Pelekis, N., and Theodoridis, Y. (2012). Easytracker: An android application for capturing mobility behavior. *2012 16th Panhellenic Conference on Informatics*, 0:357–362.

Fileto, R., Krüger, M., Pelekis, N., Theodoridis, Y., and Renso, C. (2013). Baquara: A holistic ontological framework for movement analysis using linked data. In *Conceptual Modeling*, volume 8217 of *LNCS*, pages 342–355. Springer Berlin Heidelberg.

Gabrielli, L., Rinzivillo, S., Ronzano, F., and Villatoro, D. (2013). From tweets to semantic trajectories: Mining anomalous urban mobility patterns. In Nin, J. and Villatoro, D., editors, *CitiSens*, volume 8313 of *LNCS*, pages 26–35. Springer.

Kisilevich, S., Krstajic, M., Keim, D., Andrienko, N., and Andrienko, G. (2010). Event-based analysis of people's activities and behavior using flickr and panoramio geotagged photo collections. In *14th Intl. Conf. on Information Visualisation*, pages 289–296, Washington, DC, USA. IEEE Computer Society.

May, C. and Fileto, R. (2014). Connecting textually annotated movement data with linked data. In *IX Regional School on Databases*, ERBD, São Francisco do Sul, SC, Brazil (in Portuguese). SBC.

Parent, C., Spaccapietra, S., Renso, C., Andrienko, G. L., Andrienko, N. V., Bogorny, V., Damiani, M. L., Gkoulalas-Divanis, A., de Macêdo, J. A. F., Pelekis, N., Theodoridis, Y., and Yan, Z. (2013). Semantic trajectories modeling and analysis. *ACM Computing Surveys*, 45(4):42.

Pelekis, N. and Theodoridis, Y. (2014). *Mobility Data Management and Exploration*. Springer.

Rakthanmanon, T., Campana, B., Mueen, A., Batista, G., Westover, B., Zhu, Q., Zakaria, J., and Keogh, E. (2013). Addressing big data time series: Mining trillions of time series subsequences under dynamic time warping. *ACM Trans. Knowl. Discov. Data*, 7(3):10:1–10:31.

Rinzivillo, S., de Lucca Siqueira, F., Gabrielli, L., Renso, C., and Bogorny, V. (2013). Where have you been today? annotating trajectories with daytag. In *SSTD*, pages 467–471.

Spaccapietra, S., Parent, C., Damiani, M. L., de Macedo, J. A., Porto, F., and Vangenot, C. (2008). A conceptual view on trajectories. volume 65, pages 126–146, Amsterdam, The Netherlands, The Netherlands. Elsevier Science Publishers B. V.

Wakamiya, S., Lee, R., and Sumiya, K. (2012). Crowd-sourced urban life monitoring: urban area characterization based crowd behavioral patterns from twitter. In *ICUIMC*, page 26.

Xiu-li, Z. and Wei-xiang, X. (2009). A clustering-based approach for discovering interesting places in a single trajectory. In *Intelligent Computation Technology and Automation, 2009. ICICTA '09. 2nd Int. Conf. on*, volume 3, pages 429–432.

Yan, Z., Chakraborty, D., Parent, C., Spaccapietra, S., and Aberer, K. (2013). Semantic trajectories: Mobility data computation and annotation. *ACM TIST*, 4(3).

Yin, Z., Cao, L., Han, J., Luo, J., and Huang, T. S. (2011). Diversified trajectory pattern ranking in geo-tagged social media. In *SDM*, pages 980–991. SIAM / Omnipress.

Zigkolis, C., Papadopoulos, S., Kompatsiaris, Y., and Vakali, A. (2011). Detecting the long-tail of points of interest in tagged photo collections. In Martinez, J. M., editor, *CBMI*, pages 235–240. IEEE.

Descriptive Modeling of the Web Mapping Systems Users' Behavior

Vinícius G. Braga[1], **Welder Oliveira**[1],
Vagner Sacramento[1], **Kleber V. Cardoso**[1]

[1]Federal University of Goiás (UFG)
Institute of Informatics (INF)
Goiânia – Brazil

{viniciusgoncalves, kleber, vagner}@inf.ufg.br

welder.oliveira@gogeo.io

Abstract. *Web mapping systems and Geographic Information Systems (GIS) in general are widely used nowadays. However, there is a lack of models that describe how users interact with this kind of application and the workload imposed on servers. These models are important for performance evaluation and to direct the focus of the development to the points that will affect the most used operations and the most users of the system. In this paper, we propose a descriptive model of the web mapping systems users' behavior. The model can be used as a starting point for the creation of workload generators to make performance evaluations in this kind of application. As a case study we created a workload generator and we applied it in a web tile server. We show that adding user features can significantly change the workload imposed on the server.*

1. Introduction

Web mapping systems, such as Google Maps and Bing Maps, are increasingly becoming part of people's lives and are used for different purposes. Besides these, there are several other applications that use Location Intelligence and Location Based Services in logistics, Business Intelligence and CRM systems. They are developed using either commercial (e.g. ArcGIS) or open source GIS platforms (e.g. PostGIS/GeoServer). A critical problem in GIS applications development is to ensure performance and scalability as the number of users and the data volume increase. The main factors that affect the performance of a system are the design, the implementation and the workload to which the system is subjected [Feitelson 2014]. The first two factors are well known by developers, but the last one is sometimes neglected.

Usually, performance of systems are evaluated using stress benchmarks [MAPLARGE 2014, OSGeo Wiki 2011], which are proper to help to detect bottlenecks and to compare a system with others. However, to analyze the performance of a system only using stress benchmarks is not a good choice. This kind of benchmark is very different from the workload of a system in production. They do not show the features of the system that will be most used in production and do not give a clear idea of what to priorize in the system's development.

Knowledge of the real workload allows a better resources allocation and a better cost-benefit in systems development. The developers can direct the improvement focus to

the points that will affect the most used operations and the most users. Without the correct focus, a lot of effort can be applied to improve functionalities that will be underused and vice versa. To know the workload is also important to investigate better caching strategies [Romoser et al. 2012].

Despite the great importance of knowing the real workload, there is a lack of studies in the literature on GIS user behavior and their impact on the workload imposed on servers. Some works try to simulate a real GIS workload and its effects on geographical databases [Ray et al. 2011, Simion et al. 2012]. Some works use the history to the server to improve the server cache strategies [García et al. 2012, Quinn and Gahegan 2010]. Romoser et al. (2012) created a workload characterization of the USGS EROS system, a system that allows its users navigate and download images of the Earth. However, we did not find any studies that characterize the typical user behavior and that model the workload imposed to web mapping applications.

In this paper, we carry out an investigation about Google Maps users behavior and we propose a descriptive model of the typical user behavior. The model describes several user behavior aspects, such as the distribution of the intervals between actions, the zoom levels frequency distribution, the distribution of session duration, and the frequency of the most used operations. The model also describes how users pan on a map and how the number of tile requests are intrinsically correlated with the screen resolution. In order to evaluate the impact of the model in a real system, we have created a synthetic workload and we compared it with a workload used by industry. The main contribution of this work is the description of a model that can be used as a starting point to create a workload generator to simulate real user behavior on web mapping applications.

This work is organized as follows. In Section 2, we present a brief description of web mapping systems and we talk about related works. In Section 3, we describe how the data were collected and analysed. In Section 4, we discuss results from data analysis and we present a descriptive model, which characterizes the users in several aspects. In Section 5, we show a case study with a workload generator based on the results of the data analysis and applied it to a web tile server. In Section 6, we present the final comments and we introduce some perspectives for future work.

2. Background and Related Work

Web mapping systems renderize the map in a set of fixed scales and divide it in images of the same size, called tiles. The number of tiles grows exponentially as the zoom level (scale) increases, following the 4^{zoom} rule. Google Maps, for example, works with tiles of 256x256 pixels. The number of tiles at zoom level 0 is 4^0, that is, just one tile. At zoom level 1, it is 4^1, that is, 4 tiles and so on, as can be seen in Figure 1, which shows the number of tiles increasing up to the zoom level 2. The highest zoom level that the application permits is level 21, in which the map has a resolution of 256x256 x 4^{21} pixels. This model has become a standard for web maps. It allows users request only the images related to the map part they want to see.

Despite the large use of web mapping systems, few research works seek to create a real workload model to this kind of system. Romoser et al. (2012) analyzed the logs of the USGS EROS system, a system that let users navigate and download images of the Earth, seeking to study the workload imposed to this system. They investigated the

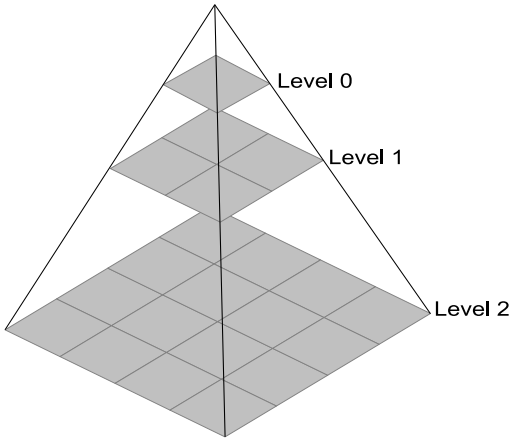


Figure 1. Tile pyramid with three levels.

accesses to the system and characterized them by user, by images and by requests. They found the most requests came from a little set of users, as well as these requests are made to a little set of popular scenes (specific places of the globe). Romoser et al. (2012) found a pattern of access to images in big disasters dates, such as earthquakes and tsunamis. These patterns were used to improve the cache and prefetching techniques, looking for improvements on system performance.

Kang et al. (2001) created an algorithm to perform prefetching of the most probable tiles to be requested based on the updated global access pattern. Furthermore, they created an algorithm to substitute the tiles in cache based on the same probabilities. The authors did not present any study of real data access on web GIS systems to prove the effectiveness of their algorithms. Other authors have implemented a prefetching strategy based on the history of previous accesses, identifying the most accessed areas and storing them in cache [Kefaloukos et al. 2012, García et al. 2012].

Ray et al. (2011) present Jackpine, a tool to benchmark spatial databases. The authors modeled several common access scenarios to these databases to simulate a real workload. Although they address several spatial database operations, the workload generator was created without taking into account several real GIS access aspects, such as think time, that is the time users take to analyze the map and perform the next action. In a later work [Simion et al. 2012], Jackpine was used to produce concurrent accesses to categorize different microbenchmarks based on CPU and disk usage. However, the workloads used by them were defined with no statistical analysis of real data, they used only good sense and the authors' experience in the industry. Whithout a statistical analysis there is no way to say that their workloads are close enough to real GIS workloads.

Zhang et al. (2007) analyzed traces of heterogeneous applications to characterize the workload imposed to the servers. They modeled the CPU usage statistically and created strategies of anomaly detection and capacity planning. Cibulka D. (2013) developed a study about the influence of the latitude, longitude and scale in the response time to recover a tile in web mapping systems. The results showed that the response time is bigger in regions with a greater density of objects, although he did not mention the cache effects.

3. Conceptualization and Methodology

Geographic Information Systems let users reference information with their locations on a map and visualize them in different types of maps. GIS allow the overlay of different georeferenced data layers that can be useful for many studies. An important use of this technology is in the creation of web mapping systems. These systems are popular and so simple to use that users do not need to have GIS knowledge to work on them.

Google Maps is one of the most popular web mapping systems of the world. Some of its operations, such as zoom and pan, are common to any web mapping system. The knowledge of the use of these operations can be useful in the characterization and modeling of the typical workload imposed to this kind of system. The latest Google Maps version, available for all users since February 2014, presents several useful information in its URL. It is possible, for example, to extract the central coordinate, the zoom level, the searches, routes, among other information. Google Maps changes the URL as a result of user actions. For example, if user moves the map, the central coordinate value is changed. This allows a reconstruction of the users' actions from the set of URLs of a Google Maps' navigation session. These points were crucial for the choice of this system in this work.

We have done a collect of a large set of accesses to Google Maps, collecting URLs and other information. With these data we have made several analysis and we characterized the users on several aspects. This section presents basic concepts about Google Maps and its URLs pattern and describes the collect and data analysis methodologies.

3.1. Google Maps

Google Maps is a popular map system and has a web and a mobile version. In this study, we focused on the web version. This version updates the URL as a result of user actions and the URLs have a lot of useful information that describe these actions. This allows the user actions to be reconstructed from the set of URLs of a navigation session. The two most basic information in the URLs are the geographic coordinate and the zoom level. Below is an example of a URL with both this information highlighted in bold. The first two numbers represent the latitude and longitude and the latter, with the "z", is the zoom level, which ranges from 0 to 21 on the road map.

- https://www.google.com/maps/@**37.3075744,-95.4133961,4z**

In addition to information about the coordinate and zoom level, it is possible to extract information about searches, routes, StreetView visualization, among others. Due to space restrictions, we will not present all the patterns, but all the cited information can be easily extracted with the right regular expressions.

3.2. Data Collection Methodology

First, we contacted companies and public agencies that work with web mapping systems and requested their systems access log. However, either they did not have this information or were unable to provide it for security reasons. In view of this, we decided to develop our own solution to collect data. We observed that the new Google Maps presents several useful information in its URL that can be used to identify user behavior features.

We developed a Google Chrome extension to collect these URLs and other information. The extension collects the Google Maps URLs and some mouse movements

inside the page, all with its respective timestamps. Additionally, it collects the page resolution in pixels, which is equivalent to the width and height of the part of the map that appears on the screen. The page resolution is collected for every user action, because the page can be resized during the session. The extension was published on Chrome Web Store and with about 120 users we have collected more than 60,000 URLs, together with other access information.

We defined a user navigation session as the time the user spends on Google Maps. This time starts when the user accesses the Google Maps page and ends when he finishes his work, either by closing the tab/window or accessing another address in the same tab. During a navigation session on Google Maps, the extension collects data on user actions. Once the user logs out, the information is assembled and sent to a server.

3.3. Data Analysis Methodology

Initially, we analyzed the timestamps of each user action and from them we identified the intervals between actions. To calculate them we created an algorithm that iterates on a set of URLs from a user session and performs the subtraction $T_{i+1} - T_i$, where T_{i+1} and T_i are the timestamps of URL_{i+1} and URL_i, respectively. The intervals represent the user think time, in other words, the time it takes to analyze the map and perform the next action. Figure 2 shows the collection time of the timestamps and what we consider to be a think time. The time T_1 was collected at the end of User Action 1, the time T_2 at the end of User Action 2 and so on. To obtain the Interval 1, we just have to subtract T_1 from T_2.

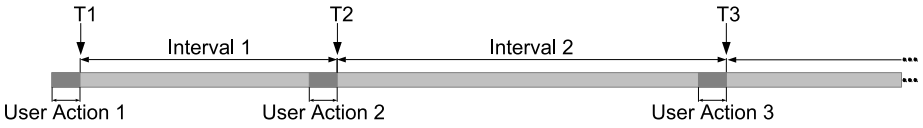


Figure 2. Interval's calculation.

Using the URLs we extracted some important information. First, we extracted the zoom level on the road map, which is the default map. The zoom level ranges from 0 to 21, and a greater zoom level means a greater scale. With this information, we were able to find the frequency of each zoom level, as shown in Figure 3(a). On the hybrid map (the map with satellite images), the zoom level is replaced with a distance information. This information varies a lot, and because of this, it were not used in the zoom level analysis.

We also analyzed the URLs in pairs to identify the action performed by the user. For that, we developed an algorithm that iterates over each set of URLs, use the Algorithm 1 to identify the actions and create an ordered list with the actions of a navigation session. Algorithm 1 analyzes two URLs in sequence and checks the change between URL_i and URL_{i+1} to define what kind of operation was performed by the user. The analysis of the first URL of a navigation session is a special case, because this URL has no predecessor, so the $urlI$ parameter is passed with a $NULL$ value and the operation is identified as $BEGIN$ type. To illustrate, suppose URLs 1 and 2 represent two sequential accesses of the same user. As can be seen, there was a change in the zoom level, passing from 4z to 5z. This change is then recorded as a zoom operation.

1. www.google.com/maps/@37.0625,-95.677068,**4z**
2. www.google.com/maps/@37.0625,-95.677068,**5z**

In many cases the URL changes in more than one part. However, different changes occur as a result of just one action. For example, let URLs 3 and 4 be two sequential accesses of the same user. As is highlighted, the URL has changed both the zoom level and central coordinate. Nevertheless, the action is recorded as a zoom, because the change in the central coordinate was due to the zoom, which is directed by the mouse pointer position when the *mousewheel* event occurs, not because of the pan. In Algorithm 1, one can see that the evaluation of zoom level change (line 8) occurs before the evaluation of the change of the central geographic coordinate (line 10), which means that the counting is done correctly. During the search and route operations, the central coordinate and the zoom level can also be changed. For this reason, Algorithm 1 evaluates changes in the route and search at first, detecting the performed operation correctly.

3. www.google.com/maps/@**37.0625,-95.677068,4z**
4. www.google.com/maps/@**38.2971386,-65.7063659,5z**

As shown above, search and route operation are recorded when a change occurs in the URL part that refers to them. For example, let URLs 5, 6 and 7 be sequential accesses of the same user. When the user searches for **drugstore**, the URL 5 changes to include the search information, as can be seen in bold in URL 6. This change causes the action to be accounted as a search operation. If the user performs a pan, the URL will continue with the search information, as can be seen in URL 7. However, as there was no change in the search part, the verification in line 4 of Algorithm 1 will return *false* and the operation will be correctly recorded as a pan.

5. .../maps/@37.0625,-95.677068,4z
6. .../maps/**search/drugstore/**@37.0625,-95.677068,4z/...
7. .../maps/**search/drugstore/@40.2194479,-80.7356618**,4z

Algorithm 1 Checks what changed between URL_i ($urlI$) and URL_{i+1} ($urlIPlus1$) and identifies the operation based on this change.

Input: $urlI$ and $urlIPlus1$
Output: The $operation$ the user carried out.

1: $operation$;
2: **if** $urlI = NULL$ **then**
3: $operation \leftarrow new\ Operation(Type.BEGIN)$;
4: **else if** $searchChanged(urlI, urlIPlus1)$ **then**
5: $operation \leftarrow new\ Operation(Type.SEARCH)$;
6: **else if** $routeChanged(urlI, urlIPlus1)$ **then**
7: $operation \leftarrow new\ Operation(Type.ROUTE)$;
8: **else if** $zoomChanged(urlI, urlIPlus1)$ **then**
9: $operation \leftarrow new\ Operation(Type.ZOOM)$;
10: **else if** $coordinateChanged(urlI, urlIPlus1)$ **then**
11: $operation \leftarrow new\ Operation(Type.PAN)$;
12: **else**
13: $operation \leftarrow new\ Operation(Type.ANOTHER)$;
14: **end if**
15: **return** $operation$;

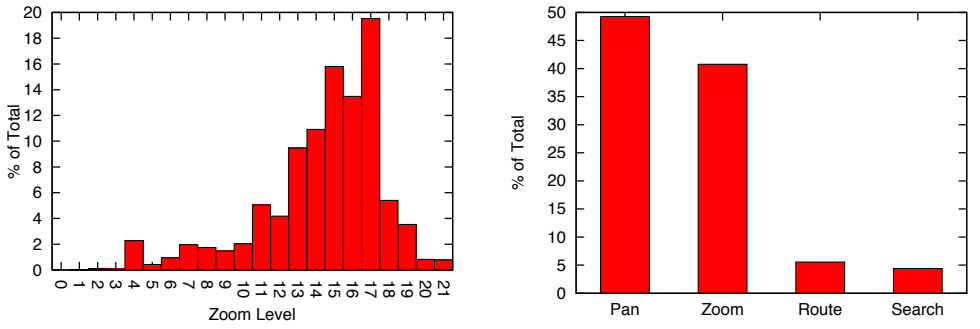
By using a coordinate and a zoom level it is possible to calculate the pixel and the corresponding tile on the map, as can be seen in the example of Google Maps API [Google Maps API 2014]. We implemented this algorithm to identify the tiles visualized on the users' screen. Each URL of the road map has the central coordinate and zoom level of the visualized map. With both these information, we calculated the central pixel position and using the page resolution we found the Bounding Box of the user screen.

We also collected the mouse positions during drag movements so that we could compute the pans sizes, but due to inertia effect on map, this information could not be used with confidence. Hence, we used the central geographic coordinate and the zoom level information of the road map's URLs to make this calculation. In every identified pan operation we computed the size of the central pixel coordinate movement using the Google Maps algorithm [Google Maps API 2014]. The distance the central pixel moves is the same distance that all map moves, which means the computed information is the pan size.

4. Descriptive Modeling

In this section, we present the results of the data analysis. All the results reveal some important user behavior characteristic, which can be used to create models and workload generators that are closer to the normal user behavior. The value of this kind of tool to systems development is described very well in Dror G. Feitelson's book [Feitelson 2014].

Figure 3(a) shows a histogram with the frequency of use of each zoom level. As can be seen, some zoom levels are used more than others and there is a greater concentration aroud zoom levels 15, 16 and 17. Zoom level 17 is the most widely used because Google Maps redirects the map automatically to this level as a result of a specific search at city level and zoom 17 allows a good navigation with a high level of details. Zoom levels greater than 17 are very close to the Earth and make the navigation difficult, which explains its low use. Very small zoom levels have a low level of details and do not bring value to most searches, that are for information and places at city level. Zoom levels closer to, but lower than, zoom 17 have a good level of details too and allow a good navigation, which explains their high use. A similar behavior was found by Garcia et al. (2012) when they analyzed traces of map servers.



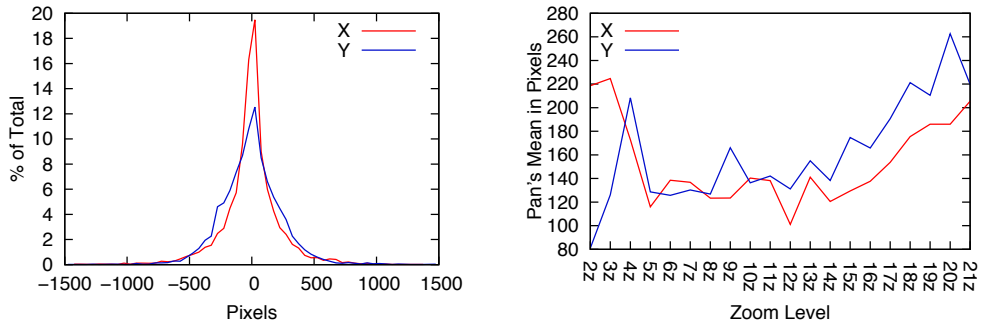
(a) Histogram of the use of zoom levels.

(b) Frequency of pan, zoom, route and search operations.

Figure 3. Evaluation of use of zoom levels and of the frequency of the four most widely used operations (b).

We made a comparison between the frequency of the most common operations performed on Google Maps. We considered the four most common operations: pan, zoom, search and route. Figure 3(b) shows the frequency of each one. Each stage of the route was treated as a route operation, since it entails spending resources on the server. As expected, pan and zoom occur with a higher frequency and a workload generator must take these frequencies into account.

However, it is not enough just to know that pans and zooms are the most frequently performed operations. We have to know how these operation are carried out, as well as, how users normally pan and zoom on a map. In view of this, we decided to conduct other analyzes. Figure 4(a) shows a histogram of the pans sizes, in pixels. The negative values represent the pans to the left and to the top on the x and y-axes, respectively, and the positive values to the right (x-axis) and to the bottom (y-axis). The number of pans are the same in both axes. As can be seen, most pans are small and, in general, users tend to make larger moves on the y-axis. This might be due to the widescreen format, which provides less information on the y-axis and results in a need for larger moves in this direction. As the widescreen format is currently the most used [StatCounter Global Stats 2014], most users move the map in this way. Figure 4(b) shows the average of the moves on the x and y-axes per zoom level and illustrates more clearly that the moves on y-axis are larger.



(a) Histogram of the pans size, in pixels.

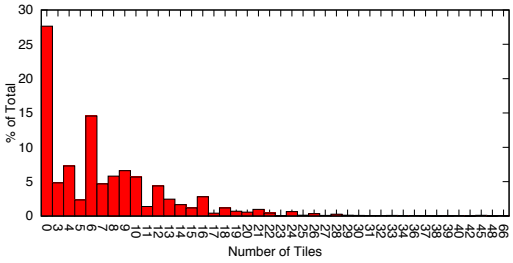
(b) Average size of pans, in pixels, per zoom level.

Figure 4. Evaluation of pan operation.

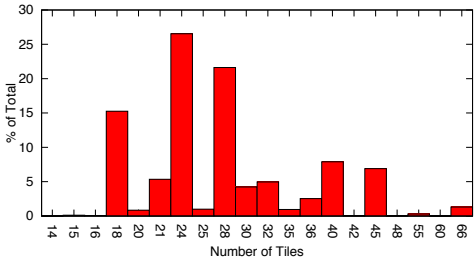
Pans and zooms were analyzed in terms of number of tiles requested by each operation and the results are shown below. Figure 5(a) shows a histogram of the number of tiles requested per pan operation. In most cases, the number of tiles is zero and this happens because of the large number of small pans that just show parts of the tiles that were already loaded and did not appear on the screen. Owing to the screen resolutions, that support at least three tiles on each axis, no pan generates a requisition of just 1 or 2 tiles. Currently, the most widely used screen resolution are 1366x768 [StatCounter Global Stats 2014], which allows a visualization of 6 tiles on the x-axis. This might explain the large number of pans that request 6 tiles, since the largest pans are made on the y-axis.

Figure 5(b) shows the histogram of the tiles requested per zoom operation. In this case, the number of tiles is directly proportional to the screen resolution. As can be seen, the top most number of tiles requested are 18, 24 and 28 and this can be justified by the fact that the most commonly used screen resolution is 1366x768. Depending on how tiles

are arranged, this resolution can fit 6 or 7 tiles on the x-axis and 3 or 4 tiles on the y-axis. If we multiply these values, we have the values 18, 24, 28 and 21, which is one of the highest values too. Frequencies of 40 and 45 tiles requested are greater than 21 and this is because that values are equivalent to the number of tiles requested on the second most widely used resolution, 1920x1080. In this resolution, the clients can request 32, 36, 40 or 45 tiles per zoom.



(a) Histogram of the number of tiles requested per pan operation.



(b) Histogram of the number of tiles requested per zoom operation.

Figure 5. Number of tiles requested per pan (a) and per zoom (b) operation.

So far, we have discused how users perform zoom and pan operations, but there is other important information which is when and for how long. Figure 6(a) shows the PDF of the intervals between user actions and Table 1 displays the mean and some important percentiles of these intervals. Although there are long intervals, some longer than 1 minute, they are not representative. As can be seen in Table 1, more than 95% of the intervals are shorter than 15 seconds. This information is very important to simulate the think time of the users and it was employed in our case study, outlined in Section 5. Figure 6(b) shows the CDF of the durations of the sessions. We made two analyzes: one directly with the collected data (Normal) and another where we only took account of sessions where there was no interval between actions greater than 1 minute (1 Min Interval's Cut). We carried out this second analysis to remove sessions in which the user leaves Google Maps open and starts to do other things. Although this kind of session has a long duration, it does not have an engaged user and there are no requests for long periods. Sessions with intervals shorter than one minute are smaller in average, but have more engaged users that are responsible for the most load on the servers.

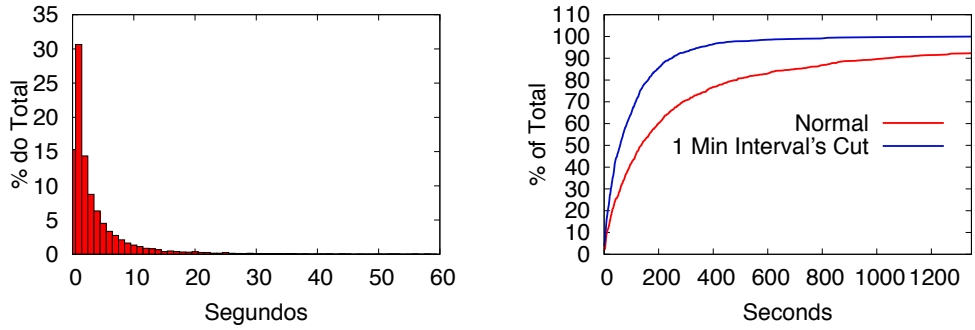
Mean	25th Percentil	50th Percentil (Median)	75th Percentil	95th Percentil
4297 ms	1302 ms	2320 ms	4608 ms	14347.1 ms

Table 1. Mean and percentiles of the intervals between user actions.

All the operations shown here are common to any web mapping system. Although the distributions might be different in other systems, the characteristics are common and the model can be extended to any web mapping system. This model can be used as a starting point to create a workload generator and simulate basic real user features, such as pans and zooms operations, the think time, among others.

5. Case Study With goGeo

Stress benchmarks are good to compare the performance between systems and to understand their behavior in stress situations. However, they do not give a clear idea of how



(a) Histogram of the intervals between actions. (b) CDF of the user sessions durations.

Figure 6. Evaluation of the intervals (a) and of the sessions durations (b).

real users will interact with the system. Without the right workload developers of a system cannot know how many real users the system can support. For this reason, there is a need to know the real workload. Since the real workload is unknown until the system goes into production, use a synthetic workload closer to the real is the best option. In Session 4, we described some characteristics of the behavior of web mapping systems users and we argued about the importance of knowing these characteristics to create workload generators. For this session, we used one of that characteristics, the intervals between actions, to create a workload closer to the real and to compare it with a stress workload. Adding the intervals, we have a workload that simulates the users' think time and that gives us a better idea of how frequently users send requests to the server.

We used the goGeo's[3] infrastructure and its web tile server to execute the tests. All tests were done with the same machines and using a 151.3 MB database of over 500,000 companies in Brazil. The set of requested tiles covers the entire Brazilian territory. The server cache was disabled to avoid interfering in the results, so that all requests require a new tile rendering. Each test ran for a period of 10 minutes and we measured the response times of each request to compare the results of the workloads.

First, we executed a stress workload based on the MapLarge performance tests [MAPLARGE 2014]. We set 25 external clients to send requests for random tiles as quickly as possible, without intervals between subsequent calls. We computed the average response time of this workload, which was 105 ms. After that, we executed some workloads with the users' think time, starting with 25 clients and increasing this number until it reached a similar average response time of the stress workload. We used an empirical distribution, based on the distribution shown in Figure 6(a), to simulate the users' think time. With 800 clients the average response time was even smaller (71 ms), but with 1,000 clients it was greater (158 ms). This means that, with a similar average response time, the system can support a number of clients between 800 and 1,000 when the intervals are taken into account.

The difference between the stress workload and the user based workload was not just in the number of clients. When we added the intervals we observed a significant increase in the autocorrelation of the response times. Figure 7 shows the autocorrelation

[3]www.gogeo.io

function (ACF) of the workloads[4]. As can be seen, there is a reasonable difference between them. While the stress workload shows autocorrelation until a time lag of 6×10^1, the user based workload reach a lag of 2.5×10^2 and 1.8×10^3 for 800 and 1,000 clients, repectively. A workload that achieves an average response time similar to the stress workload would reach a time lag between these last two values. This means that, although the average response times are the same, the time lag of the user based workload is about one order of magnitude greater than the lag of the stress workload. In this case, the higher lag in response times is related to a higher lag in the users' requests, once the set of called tiles are the same in all the tests.

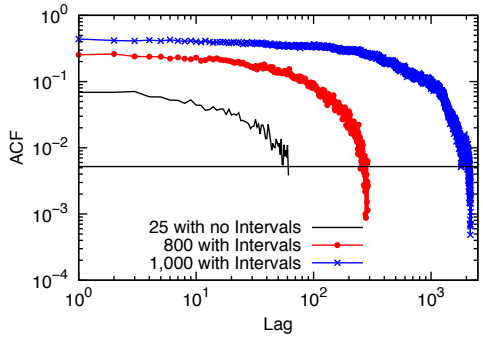


Figure 7. Evaluation of the autocorrelation.

Although the user based workload generator implemented here does not have all the characteristics of real users, it demonstrates what we want to show: adding real user features can significantly change the workload imposed on the server.

6. Conclusion and Future Work

The use of geographic information systems has increased in recent years, but there is still a lack of models for describing user behavior in this kind of application. The lack of such models makes it difficult to carry out a performance analysis and to estimate the capacity of new systems. Without a suitable workload, developers can end up by spending too much effort on system features that are rarely used and vice versa. In this study, we proposed a descriptive model of the behavior of web mapping applications users and we implemented a workload generator taking into account one of the users characteristics, the think time, to compare the results with the results of a stress benchmark. We showed that adding user features can significantly change the workload imposed on the server. In future work, we intend to improve the model to include the relationships between user actions and improve the workload generator to make the load closer to the real. Finally, we hope we have succeeded in drawing the academic community's attention to the importance of workload modeling for mapping and GIS applications in general.

References

Cibulka, D. (2013). Performance testing of web map services tn three dimensions–x, y, scale. *Slovak Journal of Civil Engineering*.

[4]The ACF is presented with confidence bounds at a significance level of 0.05, shown as a straight line parallel to the x-axis.

Feitelson, D. G. (2014). Workload modeling for computer systems performance evaluation. http://www.cs.huji.ac.il/~feit/wlmod/.

García, R., de Castro, J. P., Verdú, E., Verdú, M. J., and Regueras, L. M. (2012). Web Map Tile Services for Spatial Data Infrastructures: Management and Optimization. *Cartography - A Tool for Spatial Analysis*.

Google Maps API (2014). Showing pixel and tile coordinates. https://developers.google.com/maps/documentation/javascript/examples/map-coordinates. [Online; accessed 16-July-2014].

Kang, Y.-K., Kim, K.-C., and Kim, Y.-S. (2001). Probability-Based Tile Pre-fetching and Cache Replacement Algorithms for Web Geographical Information Systems. In *Proceedings of the 5th East European Conference on Advances in Databases and Information Systems*.

Kefaloukos, P. K., Vaz Salles, M., and Zachariasen, M. (2012). TileHeat: A framework for tile selection. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*.

MAPLARGE (2014). Map server performance. http://maplarge.com/mapserverperformance. [Online; accessed 16-July-2014].

OSGeo Wiki (2011). Benchmarking 2011. http://wiki.osgeo.org/wiki/Benchmarking_2011. [Último acesso: 31-Julho-2014].

Quinn, S. and Gahegan, M. (2010). A predictive model for frequently viewed tiles in a web map. *Transactions in GIS*.

Ray, S., Simion, B., and Brown, A. D. (2011). Jackpine: A benchmark to evaluate spatial database performance. In *Data Engineering (ICDE), 2011 IEEE 27th International Conference on*.

Romoser, B., Fares, R., Janovics, P., Ruan, X., Qin, X., and Zong, Z. (2012). Global workload characterization of a large scale satellite image distribution system. In *Performance Computing and Communications Conference (IPCCC), 2012 IEEE 31st International*.

Simion, B., Ray, S., and Brown, A. D. (2012). Surveying the Landscape: An In-depth Analysis of Spatial Database Workloads. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*.

StatCounter Global Stats (2014). Top 10 Desktop Screen Resolutions on July 2014 StatCounter Global Stats. http://gs.statcounter.com/#desktop-resolution-ww-monthly-201407-201407-bar. [Online; accessed 16-July-2014].

Zhang, Q., Cherkasova, L., Mathews, G., Greene, W., and Smirni, E. (2007). R-Capriccio: A Capacity Planning and Anomaly Detection Tool for Enterprise Services with Live Workloads. In *Proceedings of the ACM/IFIP/USENIX 2007 International Conference on Middleware*.

RDebug: A New Debugging Technique for Distributed R-Trees

Sávio S. T. de Oliveira[2], Jose F. de S. Filho[1], Vagner J. do Sacramento Rodrigues[2], Marcelo de C. Cardoso[2], Sérgio T. de Carvalho[1]

[1]Instituto de Informática – Universidade Federal de Goiás (UFG)
Alameda Palmeiras, Quadra D, Câmpus Samambaia
131 - CEP 74001-970 – Goiânia – GO – Brazil

[2]GoGeo
Rua Leopoldo Bulhões, esquina com a Rua 1014
Quadra 31, Lote 07, Sala 9 Setor Pedro Ludovico
CEP 74820-270 – Goiânia – GO – Brazil

savio.teles@gogeo.io, jkairos@gmail.com, vagner@gogeo.io,
marcelo.cardoso@gogeo.io, sergio@inf.ufg.br

Abstract. *The high data availability and the increasing number of GIS users have motivated the emergence of distributed algorithms to process spatial operations efficiently. They are based on distributed indexes for an high performance processing. Researches and ongoing work use R-trees as a distributed spatial structure for indexing and retrieval of geo objects. However, these indexes have shown a challenge, that is, how to debug an index which is based on an R-Tree? In the past few years researches have been published on both distributed algorithms and distributed processing. Though none of them has proposed a debugging technique to a distributed R-Tree index. This paper presents a new algorithm for debugging a distributed index based on R-Tree which is called RDebug. This algorithm was used on DistGeo, a platform to process distributed spatial operations. A graphic tool, named RDebug Visualizer, was developed to show the output of the RDebug algorithm.*

1. Introduction

The increasing of large spatial datasets demands high performance engine in order to process complex spatial models. The best cost-benefit to provide innovative GIS applications taking advantage of all available data is through distributed and parallel GIS processing. But develop high performance engine to distributed spatial computing is very complex and challenging.

In order to handle spatial data efficiently, a database system needs an index mechanism that will help it retrieve data items quickly according to their spatial locations. The R-Tree typically is the preferred method for indexing spatial data. Many researches such as [An et al. 1999, de Oliveira et al. 2011, Zhong et al. 2012], show that a distributed index structure can provide an efficient mechanism of spatial operations processing.

However, distributed R-Trees indexes for Big Spatial Data are very complex to be developed and so it demands novel approaches to debug and check stability and this is the main issue investigated in this work.

Debugging is an essential step in the development process, though often neglected in the development of distributed applications due to the fact that distributed systems complicate the already difficult task of debugging [WH Cheung 1990]. In recent years, researches have developed some helpful debugging techniques for distributed environment. Nevertheless, we have not found in the literature any work that have addressed the problem of debugging a distributed R-Tree.

In this paper, we propose a new debugging algorithm for distributed R-Tree building. The debugging algorithm, called RDebug, uses the distributed index structure to aggregate debugging information. RDebug is used on DistGeo, a shared-nothing platform for distributed spatial algorithms processing. We have also created a graphical tool to visualize the debugging information and the R-Tree index structure, called RDebug Visualizer.

The main contributions of this paper are as follows:

- RDebug - A debugging technique for distributed R-Tree building.
- DistGeo - A peer-to-peer platform, with no single point of failure, to process distributed spatial algorithms of an R-Tree.
- RDebug Visualizer - A graphical tool to visualize debugging information and the distributed R-Tree index.

The rest of the paper is structured as follows. In Section 2, we briefly give an overview of the use of debugging techniques for distributed environments and the view of the distributed spatial algorithms. Section 3 describes the distributed processing of spatial algorithms, Section 4 presents our approach for distributed R-Tree debugging. Section 5 presents the evaluation of RDebug algorithm in the DistGeo platform. Finally, we close the paper with some concluding remarks in Section 6.

2. Related work

Researches on distributed spatial data either show techniques to debug distributed applications in general or techniques for R-tree distributed processing, but none addressed both issues. The Section 2.1 shows the distributed debugging researches and 2.2 describes researches of platforms for processing distributed spatial algorithms.

2.1. Distributed Debugging Techniques

In [G. et al. 2011] the author breaks down debuggers in two main families: log-based debuggers (also known as post-mortem debuggers) and breakpoint-based debuggers (also known as online debuggers). Log-based debuggers insert log statements in the code to be able to generate a trace log during its execution. Breakpoint-based debuggers, on the other hand, execute the program in the debug mode that allows programs to pause/resume the execution at certain points, inspect the state and the perform step-by-step execution.

Several breakpoint-based debuggers have been designed for parallel programs using message passing communication including p2d2 [Hood 1996], TotalView [Gottbrath 2009], and Amoeba [Elshoff 1989]. These debuggers offer the traditional commands to stop, inspect and execute step-by-step a running program. Some of them allow to set breakpoints on statements of one process (e.g. [Gottbrath 2009]) or a set of processes (e.g. [Hood 1996], [Elshoff 1989]).

A great body of concurrent and parallel debugging techniques are event-based. Event-based debuggers [C. E. Mcdowell 1989] conceive the execution of a program as a sequence of events. The debugger records the history of the events generated by the application, which can then be used to either browse the events once the application is finished [Fonseca et al. 2007, Stanley et al. 2009], or to replay the execution to recreate the conditions under which the bug was observed.

[WH Cheung 1990] describes a process for distributed debugging in general and does not focus on a specific debugger or a particular technique, the paper focus is on defining a step-by-step approach to tackle distributed debugging independent of the environment.

2.2. Distributed Spatial Algorithms

This Section describes briefly the researches which present the use of parallelism in order to improve the response time of the spatial algorithms. M-RTree [Koudas et al. 1996] was the first published paper, which shows a shared-nothing architecture, with a master and several workstations connected to a LAN network. The master machine can be a bottleneck because it handles and client requests and moreover merges the answers of the slaves and send to client machines. A similar technique was found on MC-RTree [Schnitzer and Leutenegger 1999] and [An et al. 1999], which show the same problems on master machine.

Hadoop-GIS [Kerr 2009] shows a scalable and high performance spatial data warehousing system for running large scale spatial queries on Hadoop. However, it does not use index to process the spatial operations. [de Oliveira et al. 2011] presents a platform to process distributed spatial operations. Although, the solution proposed in [de Oliveira et al. 2011] implements a distributed index, it is not scalable, since every message go through the replicated master node. [de Oliveira et al. 2013] shows a hybrid peer-to-peer platform, which comprehends a set of machines for naming resolution that could be a bottleneck in the system.

[Xie et al. 2008] introduces a two-phase load-balancing scheme for the parallel GIS operations in distributed environment. [Zhang et al. 2009] describes MapReduce and shows how spatial queries can be naturally expressed in this model. However, it is only indicated for non-indexed datasets.

A number of techniques and platforms have been proposed for handling spatial big data. Nevertheless, none of the researches propose a technique for distributed spatial index debugging of an R-Tree. Besides, none of them propose a platform using a peer-to-peer approach for processing distributed spatial algorithms as found on DistGeo platform (Section 3.1).

3. Distributed Processing of Spatial Algorithms

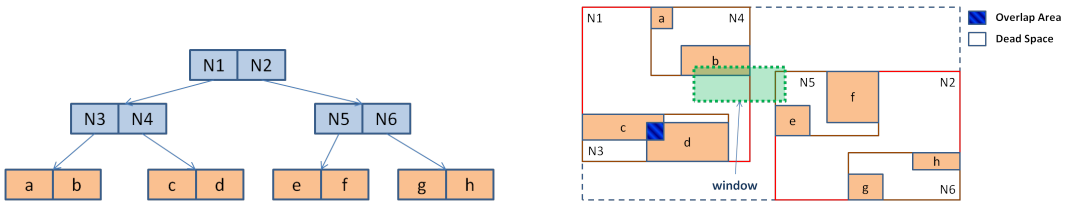
A number of structures have been proposed for handling multi-dimensional spatial data, such as: KD-Tree [Bentley 1975], Hilbert R-Tree [Kamel and Faloutsos 1994] and R-Tree [Guttman 1984]. The R-Tree has been widely used to index the datasets on GIS databases and it has been used as an index data structure in this work.

An R-Tree is a height-balanced tree similar to a B-Tree [Comer 1979] with index records in its leaf nodes containing pointers to data objects. The key idea of the data

structure is to group nearby objects and represent them with their minimum bounding rectangle (MBR) in the next higher level of the tree.

Figure 1 illustrates the hierarchical structure of an R-Tree with a root node, internal nodes ($N1...2 \subset N3...6$) and leaves ($N3...6 \subset a...h$). Every internal node contains a set of rectangles and pointers to the corresponding child node and every leaf node contains the rectangles of spatial objects.

The Figure 1(b) shows MBRs grouping spatial objects of $a...h$ in sets by their co-location. The Figure 1(a) illustrates the R-Tree representation. Each node stores at most M and at least $m \leq M/2$ entries [Guttman 1984]. Our work uses the formula for M value calculation presented in [de Oliveira et al. 2011].



(a) R-Tree index (b) Geographic space

Figure 1. R-Tree Structure

The Window Query is one of major query algorithms in R-Tree. The search starts from the root node of the tree and the input is a search rectangle (Query box). For each rectangle in a node, it has to be decided whether it overlaps the search rectangle or not. If so, the corresponding child node has to be searched too.

Searching is done recursively until all overlapping nodes have been traversed. When a leaf node is reached, the contained bounding boxes (rectangles) are tested against the search rectangle and the objects that intersects with the search rectangle are returned.

In Figure 1, the search starts on root node, where the window intersects with nodes $N1$ and $N2$. Then, the algorithm analyses node $N1$, which only $N4$ intersects with the window. Analysing node $N4$, the algorithm returns the spatial object namely $'b'$, that is the single object that intersects the window.

In node $N2$, we do not have any entry intersecting with the window due to the dead space. In other words, the window intersects with a space, which does not contain any data. The dead space should be minimized to improve the query performance, since decisions which paths have to be traversed can be taken on higher levels.

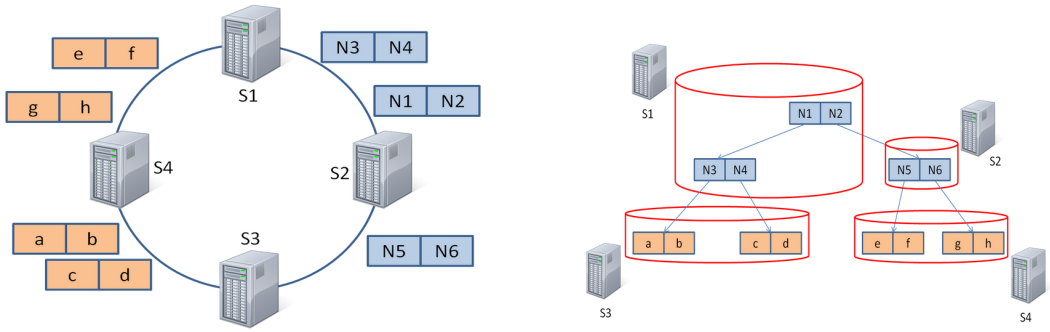
The overlapping area between rectangles should be minimized as well, as it degrades the performance of R-Tree [Beckmann et al. 1990]. Less overlapping reduces the amount of sub-trees accessed during r-tree traversal. The area between c and d in Figure 1 is an example of overlapping.

3.1. DistGeo: A Platform of Distributed Spatial Operations for Geoprocessing

DistGeo is a platform to process spatial operations in a cluster of computers (Figure 2). It is based on a shared-nothing architecture, which the nodes do not share CPU, hard disk and memory and the communication relies on message exchange. Figure 2(a) depicts

DistGeo platform based on peer-to-peer model presented as a ring topology. It is divided in ranges of keys, which are managed for each server of the cluster. In order to a server join the ring it must be assigned a range first.

The range of keys are known by each server in the cluster using a Distributed Hash Table (DHT) to store the mapping of the keys to servers. For instance, in a ring representation, whose key set start with 0 to 100, if we have 4 nodes in the cluster, the division could be done as shown below: a) 0-25, b) 25-50, c) 50-75 e d) 75-100. If we want to search for one object with key 34, we certainly should look on the server 2.



(a) DistGeo Architecture (b) R-Tree Partitioning in DistGeo

Figure 2. DistGeo Platform

Every replica of an object is equally important, in other words, there is not a master replica. Read and write operations may be performed in any server of the cluster. When a request is made to a cluster's server, it becomes the coordinator of the operation requested by the client. The coordinator works as a proxy between the client and the cluster servers.

DistGeo uses the Gossip protocol [Demers et al. 1987], which every cluster server exchanges information among themselves for everyone knows the status of each server. In the Gossip protocol every second a message is exchanged among three servers in the cluster, consequently every cluster's server have knowledge of each other.

Figure 2(b) illustrates the structure of a Distributed R-Tree in a cluster. The partitioning it is performed grouping the servers in cluster and creating the indexes according to the R-Tree structure. The lines in Figure 2(b) show the need for message exchange to reach the sub-trees during the algorithm processing.

Insertions and searching in a distributed R-Tree are similar to the non-distributed version, except for: i) The need of message exchange to access the distributed partitions and ii) Concurrency control and consistency due to the parallel processing in the cluster. Both were implemented on DistGeo platform.

The distributed index has been built according to the taxonomy defined in [An et al. 1999], as follows: i) Allocation Unit: block - A partition is created for every R-Tree node; ii) Allocation Frequency: overflow - In the insert process, new partitions are created when a node in the tree needs to split; iii) Distribution Policy: balanced - To keep the tree balanced the partitions are distributed among the cluster servers.

Reliability and fault-tolerance were implemented on DistGeo storing the R-Tree nodes in multiple servers in the cluster. The DistGeo uses Apache Cassandra [Cassandra]

database to store the distributed R-Tree index nodes on cluster servers. Each R-Tree node N receives a key, which is used to store the node in a server S responsible for ring range, replicating the node N to the next two servers in S (clockwise). If a message is sent to N, is selected one of the servers that store a replica of N. The query requests are always sent to one of the cluster's server that stores the root node of the R-tree.

As discussed on Section 3, reducing the overlapping and dead area on R-Tree minimizes the number of R-Tree nodes accessed during the tree traversal on search algorithms. The growth of the number of nodes accessed increase the network traffic because the R-Tree nodes are stored in several servers on cluster, as shown in Figure 2(b). This work implements a new algorithm that collects debugging information about a distribute R-Tree and can helps to reduce the overlapping and dead area. We cover this algorithm in more details in Section 4.

4. A Technique for Debugging A Distributed R-Tree

Guarantee that a distributed spatial index has being built accordingly is a non-trivial task. In a distributed environment, it is hard to find bugs on insertion algorithms due the difficult to synchronize the insertion, since it must be done concurrently. Even in cases where the implementation is correct, it is not easy to improve the insertion algorithm's performance (for example reducing the overlapping) due the intricacy to collect information about the spatial index.

This section describes RDebug, a new technique for index debugging, which allows collect debugging information about the distributed spatial index once it has been created. The following debug information about building consistency of the R-Tree index are collected by RDebug: i) if each R-Tree node N are consistent between the servers that store any replica of N; ii) if the MBR of each parent node intersects with the MBR of their children, iii) the presence of duplicated nodes on R-Tree or nodes being referenced by more than one parent node, and iv) if the value M and m of the nodes are compliant with the R-Tree descriptions as shown in Section 3. Furthermore, it is possible to access index data to help in optimization and minimizing the dead space and overlapping area.

The RDebug algorithm was based on R-Tree structure because it is used to index the spatial datasets on DistGeo platform presented in our work on Sub-Section 3.1. RDebug can be used with any index similar to R-Tree like Hilbert R-Tree [Kamel and Faloutsos 1994], since the RDebug algorithm uses the nodes organization of the R-Tree to collect the debug information.

Algorithm 1 shows the RDebug technique for debugging the distributed spatial index, using the index structure itself. The algorithm has two steps: 1) The algorithm processing is similar to the search in an R-Tree with a top-down traversal; 2) The algorithm does a bottom-up traversal on R-Tree, constructing the result with the debug information.

RDebug has been implemented on DistGeo platform. The R-Tree nodes are distributed and replicated over the cluster. Thus, RDebug can be processed on DistGeo platform without bottlenecks and point of failures. Besides, the R-Tree replicated nodes in the cluster allow load-balancing in the distributed R-Tree index traversal. During the traversal, at every node accessed the traversal might go to a node of the cluster with less workload, increasing the RDebug algorithm performance.

Algorithm 1: $RDebug(T)$

Data: T reference of the root node of R-Tree $tree$
Result: Debugging information about distributed R-Tree $tree$

```
1  S1 [Search subtrees]
2  if T is not leaf then
3      stores the number of children entries in each replica server of T
4      for each entry E in T do
5          server ← choose one server, randomly, that keep one replica of E
6          send msg to server to process the node's child of E on step S1
7      end
8  else
9      verify the consistency of T in other replicas
10     Invoke step S2 [Aggregation]
11 end
12 S2 [Aggregation]
13 information ⇐ the child's information stored on shared memory by
   replicas of T
14 replica_consistency ⇐ verify the consistency of T in others replicas
15 node_consistency ⇐ verify the consistency of M and m values of T
16 overlap ⇐ overlap area of T
17 dead_area ⇐ dead area of T
18 bound ⇐ MBR of T
19 add in information: replica_consistency, node_consistency, overlap,
   dead_area, bound
20 if T is leaf then
21     if T is root then
22         send response with R-Tree nodes information to app client
23     end
24     send msg with information to parent of T
25 else
26     entry_info ⇐ information sent by child node
27     mbr_consistent ⇐ verify if the bound of the child node is equal to
       bound of entry of T that points to this child
28     add in information: entries_info, and mbr_consistent
29     count ⇐ retrieve the number of child entries, which did not send a
       debugging response and decrement by 1
30     if count == 0 then
31         if T is root then
32             send response with information to client
33         else
34             send msg with information to parent of T
35         end
36     else
37         store information on shared memory
38     end
39 end
```

In the first step, called S1 [Search sub-trees] (lines 1 - 11), the Algorithm 1 traverses every node of the R-Tree starting from the root node to the leaves. The first request is sent to any server, which stores a replica of the root node.

If the node T is not a leaf (lines 2 - 8), then the number of children entries is stored to control the number of expected answers associated to T in the second step of the algorithm. This information is stored in a shared memory accessed by all servers with a replica of T. Lines $4 - 7$, show that for each entry E in T, a message is sent (continuing step S1) to any server that holds a replica of the child node of E, carrying on the first step in the children nodes. If T is a leaf, the second step, named S2 [Aggregation] is started.

Second step aims (lines $12 - 39$) to aggregate the information about the index to be used for future debugging. This step returns debugging information about each node of the R-Tree. The index itself is used to aggregate this information using the cluster computational resources to improve the algorithm's performance. The index reverse structure facilitates the collection of the debugging information, as one node of the R-Tree is responsible to aggregate only the information of its children.

The debug information about each node of R-Tree in stored in a shared memory that can be accessed by any server that store a replica of T. The RDebug update the information about the node T that is being analyzed between the lines $13 - 18$.

In the line 13, the information is retrieved from the shared memory. Line 14 verifies the consistency of T in the servers that store any replica of T. Line 15 verifies the consistency of M and m values. Lines 16 and 17 calculate the overlap and the dead space area, respectively, for each node of the R-tree. Line 18 get the MBR of the T. This information is inserted in $information$ on line 18.

If the aggregation step is being executed in the leaves (lines 20 - 24), then there are two options. If T is the root node (line 22), the node information is sent to the client application. If T is not the root node, in line 24, the information is sent to the parent node of T.

If the aggregation step is in an internal node (lines 26 - 39), the algorithm aggregates the information of the children nodes. In the line 29, the algorithm receives the information sent by the child node. Line 27, verifies if the MBR of the entry that points to the child node is indeed the same MBR sent by the child node.

Line 28 adds the data processed from lines 26 and 27 in $information$. Line 29 acquires the number of children nodes that not sent debugging information yet. This value is stored in the variable $count$, which is decremented and updated on shared memory.

If every node has sent the answer, the variable count then will hold the value 0 and lines 30-35 are processed. If T is the root node, then the information is sent to the client application, otherwise, all information collected is sent to the parent node of T. If the variable $count$ is greater than 0, then the client information is stored in the shared memory to be used until until each reply is received by child nodes.

The algorithm 1 was implemented in the DistGeo platform to collect the debugging information of the built distributed R-tree. This information is used in the platform to find out indexing issues and for speed up the searching on an R-Tree. Using RDebug algorithm it is possible debug the searching algorithms in a single R-Tree. For exam-

ple, the Window Query algorithm shown on Section 3. Whereas, algorithms that access many R-Trees, such as Spatial Join, need a deep change, as the algorithms can go through different paths.

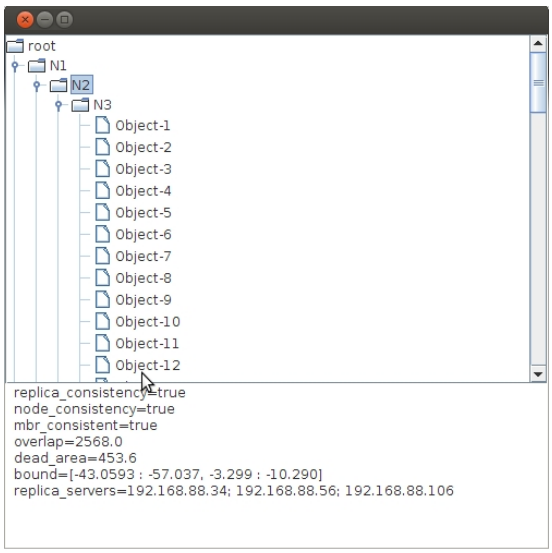


Figure 3. RDebug Visualizer

The algorithm RDebug have collected debugging information about the R-Tree index built during the insertion of the dataset. Figure 3 shows a graphical tool (RDebug Visualizer) created in our work to visualize the collected debugging information. RDebug Visualizer shows the structure of the distributed R-Tree index and allows the analysis of each node of the R-Tree. The output of the RDebug algorithm shows which nodes are currently inconsistent. The user can access the path of the node and visualize the node's inconsistent information.

5. Evaluation

The RDebug algorithm have been evaluated on 3500 MHz Intel(R) Core(TM) i7-2600 CPU workstations connected by 1 GBit/sec switched Ethernet running Ubuntu 14.04. Each node has 16 GB of main memory. The experiment results were achieved with 1, 2, 4 and 8 servers on DistGeo platform.

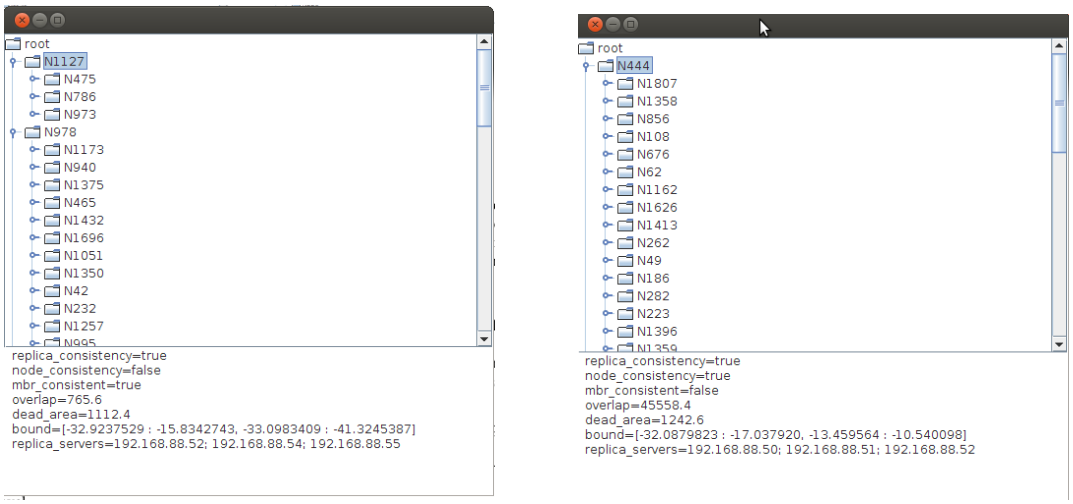
The experiments were performed using three datasets with different characteristics. The first contains 1000000 points of business listings and points of interest (POIs) from SimpleGeo[1]. The second dataset comprises 226964 lines representing the rivers on Brazil from LAPIG[2]. The third contains 220000 polygons of the census of USA from TIGER/Line[3].

The RDebug was executed on DistGeo platform after the indexing of each dataset. The algorithm was able to collect information about the R-Tree index, such as dead space

[1]https://github.com/simplegeo
[2]www.lapig.iesa.ufg.br
[3]Census 2007 Tiger/Line data

and overlapping area. Furthermore, RDebug algorithm has succeeded to collect the index structure allowing visualize each data set R-Tree index on RDebug Visualizer tool.

Three inconsistencies were deliberately inserted in the index to evaluate the RDebug: i) inconsistencies between parent and child nodes bounding, ii) nodes filled with more than M entries and iii) duplication of a node on R-Tree. The RDebug algorithm was able to identify this inconsistencies in every distributed R-Tree related to datasets. The replica consistency on DistGeo is provided by Apache Cassandra [Cassandra] and no replica inconsistencies was found in any test.



(a) Node inconsistency (b) Bound inconsistency

Figure 4. RDebug algorithm on business listings dataset

Figure 4 shows the result of RDebug algorithm with the business listings dataset in RDebug Visualizer tool. An example of node inconsistency is shown in Figure 4(a), which the R-Tree node *N1127* contains only three entries. This number of entries violates the m value presented in Section 3. Figure 4(b) shows the bound inconsistency between node *N444* and one of its children. The duplicated nodes identified on R-Tree are shown on final report by RDebug algorithm. The user can traverse the R-Tree path on RDebug Visualizer to identify these duplicated nodes.

6. Conclusion

DistGeo platform presents an approach for processing distributed spatial operations through the distributed R-Tree index. Due to the distributed processing nature on this platform an issue arises: debugging the R-Tree index distributed in a cluster of computers.

We have seen researches on spatial data processing and distributed debugging, but none of them propose techniques for debugging spatial algorithms in an R-Tree. Our work presents the RDebug algorithm for debugging the building of a distributed R-Tree index. RDebug uses the R-Tree index itself to gather the debug information. The data gathering is achieved in a distributed way, improving the debugging algorithm efficiency.

A new peer-to-peer platform (DistGeo) was proposed in our work to process distributed spatial algorithms. RDebug has been implemented in DistGeo platform. The

R-Tree nodes are distributed and replicated over the cluster. Thus, RDebug can be processed without bottlenecks and point of failures.

A graphical tool(RDebug Visualizer) has been created to visualize the structure of the distributed R-Tree index and the debugging information about the index building. Using this debugging information, we can identify discrepancies in the index building and optimize the R-Tree index too. The RDebug algorithm can be used to collected debug information in any index with spatial nodes organization similar to R-Tree (e.g. Hilbert R-Tree [Kamel and Faloutsos 1994]).

Ongoing work includes modify the RDebug algorithm to debug the Window Query and Join Query searching algorithms. The RDebug algorithm is easily adapted to gather debugging information for Window Query. Whereas, for Join Query algorithm, RDebug must be changed considerably, since the traversal is processed in two different distributed R-Trees. Another ongoing work is to simulate node replica inconsistencies to evaluate the ability of the Rdebug to identify this inconsistencies. On future works, the algorithm RDebug will be evaluated in larger clusters and performance results will be collected.

References

An, N., Lu, R., Qian, L., Sivasubramaniam, A., and Keefe, T. (1999). Storing spatial data on a network of workstations. *Cluster Computing*, 2(4):259–270.

Beckmann, N., Kriegel, H., Schneider, R., and Seeger, B. (1990). *The R*-tree: an efficient and robust access method for points and rectangles*, volume 19. ACM.

Bentley, J. (1975). Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):517.

C. E. Mcdowell, D. P. H. (1989). Debugging concurrent programs. *ACM Computing Surveys*, 21:593–622.

Cassandra, A. The apache software foundation. *URL: http://cassandra. apache. org/(visited on 24/08/2013)*.

Comer, D. (1979). Ubiquitous B-tree. *ACM Computing Surveys (CSUR)*, 11(2):121–137.

de Oliveira, S. S., Vagner, J., Cunha, A. R., Aleixo, E. L., de Oliveira, T. B., Cardoso, M. d. C., Junior, R. R., Bloco, I., and Campus, I. (2013). Processamento distribuído de operações de junção espacial com bases de dados dinâmicas para análise de informações geográficas. *XXXI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*.

de Oliveira, T., Sacramento, V., Oliveira, S., Albuquerque, P., Cardoso, M., Bloco, I., and Campus, I. (2011). DSI-Rtree - Um Índice R-Tree Escalável Distribuído. In *XXIX Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*.

Demers, A., Greene, D., Hauser, C., Irish, W., Larson, J., Shenker, S., Sturgis, H., Swinehart, D., and Terry, D. (1987). Epidemic algorithms for replicated database maintenance. In *Proceedings of the sixth annual ACM Symposium on Principles of distributed computing*, pages 1–12. ACM.

Elshoff, I. J. P. (1989). A distributed debugger for amoeba. In *In Symposium on Parallel and distributed tools*, pages 1–10. SIGPLAN Not.

Fonseca, R., Porter, G., Katz, R. H., Shenker, S., and Stoica, I. (2007). X-Trace: A pervasive network tracing framework. In *4th USENIX Symposium on Networked Systems Design And Implementation*, pages 271–284. Cambridge MA, USA.

G., B. E., V., C. T., C., N., D., M. W., and D, H. T. (2011). REME-D: a Reflective Epidemic Message-Oriented Debugger for Ambient-Oriented Applications. *ACM*, pages 1275–1281.

Gottbrath, C. (2009). Deterministically troubleshooting network applications. In *Technical report, TotalView Technologies*. TotalView Technologies.

Guttman, A. (1984). *R-trees: a dynamic index structure for spatial searching*, volume 14. ACM.

Hood, R. (1996). The p2d2 project: building a portable distributed debugger. In *In Symposium on Parallel and distributed tools*, pages 127–136. ACM.

Kamel, I. and Faloutsos, C. (1994). Hilbert R-tree: An Improved R-tree using Fractals. In *VLDB 20th*, page 509. Morgan Kaufmann Publishers Inc.

Kerr, N. (2009). Alternative Approaches to Parallel GIS Processing. *Arizona State University - Master Thesis*.

Koudas, N., Faloutsos, C., and Kamel, I. (1996). Declustering spatial databases on a multi-computer architecture. *Advances in Database Technology-EDBT'96*, pages 592–614.

Schnitzer, B. and Leutenegger, S. (1999). Master-client r-trees: A new parallel r-tree architecture. In *Scientific and Statistical Database Management, 1999. Eleventh International Conference on*, pages 68–77. IEEE.

Stanley, T., Close, T., and Miller, M. (2009). Causeway: A message-oriented distributed debugger. In *Technical Report HPL-2009-78*. HP Laboratories.

WH Cheung, JP Black, E. M. (1990). A Framework for Distributed Debugging. *IEEE*, pages 106–115.

Xie, Z., Ye, Z., and Wu, L. (2008). A two-phase load-balancing framework of parallel gis operations. In *Geoscience and Remote Sensing Symposium, 2008. IGARSS 2008. IEEE International*, volume 2, pages II–1286. IEEE.

Zhang, S., Han, J., Liu, Z., Wang, K., and Feng, S. (2009). Spatial queries evaluation with mapreduce. In *Grid and Cooperative Computing, 2009. GCC'09. Eighth International Conference on*, pages 287–292. IEEE.

Zhong, Y., Han, J., Zhang, T., Li, Z., Fang, J., and Chen, G. (2012). Towards parallel spatial query processing for big spatial data. In *Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW), 2012 IEEE 26th International*, pages 2085–2094. IEEE.

Prediction of cattle density and location at the frontier of Brazil and Paraguay using remote sensing

Thaís Basso Amaral[1], Valery Gond[2], Annelise Tran[3]

[1]Embrapa Gado de Corte – Av. Rádio Maia nº 830, Zona Rural - 79106-550 - Campo
Grande - MS - Brazil

[2]CIRAD, UPR 105 Biens et services des écosystèmes forestiers tropicaux
Montpellier - France

[3]CIRAD, UPR AGIRs "Animal et Gestion Intégrée des Risques", Maison de la télédétection - Montpellier - France

thais.amaral@embrapa.br, valery.gond@cirad.fr, tran@teledetection.fr

Abstract

In this paper, we explore the potential of remote sensing to map pastures areas and by this way establish models for predicting cattle density and location. First, an object based classification (OB) was made in Landsat 5 images for three different municipalities to provide a land-cover map. Second, on the basis of Brazilian official livestock database, a statistical model to predict number of cattle in function of declared pasture area by the farmers was produced. Finally, this model was applied to the pasture areas detected by remote sensing to predict cattle density. Coefficient of determination of the model was 0.63. The results indicate that the methodology used for estimating cattle density has a potential to be applied in regions where no information about farm location and cattle density exists.

Key-words: cattle density, landsat 5 images, linear regression, object based classification, pasture areas.

1.Introduction

The meat production chain is the largest generator of jobs in Mato Grosso do Sul (MS) State, having the third largest beef herd in Brazil (22 million heads) according to the Brazilian Ministry of Agriculture, Livestock and Supply (2010). It has also the largest network of qualified exporters' slaughterhouses to the European Union. This, associated with the fact of having 1.131 km of international border, makes the State strategic for the control and prevention of diseases that are included in World Organization for Animal Health (OIE) list and affects international trades such as Foot and Mouth Disease (FMD).

As the frontier with Paraguay extends over a range around 1.100 km, there is a considerable change of landscape, soil and microclimates that leads to changes in agricultural practices and production systems. These different types of production systems are not very well documented and they play a significant role in terms of risk of the spread of FMD virus. Moreover, in Brazil farm cadastre is not georeferenced. By this means it is important to find another mean to characterize and spatialize cattle production systems in the frontier. By this reason, remote sensing could be potentially used to map and to monitor pastures areas and by this way establish models for predicting cattle density and location.

Maps of cattle density provided by FAO (2005) were produced from census livestock statistics but in a broad scale, not precising location of pasture areas inside the municipality and the estimative are more generalized for the municipalities. The objective of this paper is to propose a methodology to predict cattle density and location from medium resolution satellite images using image oriented object classification.

2.Methods

2.1.Study area

The studied region comprehended three sites (municipalities) located in the frontier between Brazil and Paraguay in Mato Grosso do Sul State. The municipalities were chosen because they represent the differences that exists in the frontier and they are also well geographically distributed, Porto Murtinho in the extreme north, Ponta Porã in the center and Mundo Novo in the south (Figure 1).

Figure1. Map of Mato Grosso do Sul State, with the localization of the study areas (Porto Murtinho, Ponta Porã and Mundo Novo)

Mato Grosso do Sul State climate is classified, according to Köppen, as tropical-type Aw, with an average temperature of 24.4 °C in the warmer months (January and February) and 19.1 °C during the coldest months (June and July). The average annual precipitation is 1.470 mm. January is the wettest month (average of 243 mm of rain and 81% relative humidity) and August the driest (40 mm of rain and 60% relative humidity, on average).

Mundo Novo municipality is the smallest studied area, occupying an area of 480 km^2 with approximately 30,000 head of cattle, characterized by small producers, who mainly develop dairy farming and subsistence agriculture. Ponta Porã is potentially an agricultural region, with 5,328 km^2, where co-exist large farms and family farming, with around 3,000 properties of smallholder farming systems and approximately 231,000 cattle heads. Porto Murtinho is the largest municipality 17,734 km , mainly characterized by extensive livestock production, with a predominance of medium and large beef cattle farms with a herd of approximately 650,000 bovines.

2.2. Remote sensing data

Four Landsat 5 TM images from August and September 2010 with 30-meter spatial resolution were downloaded from Brazilian Institute for Space Research (INPE, www.inpe.br). Pre-processing included geometric correction, mosaicing and the creation of subsets for the study area. Geometric rectification of the imagery was undertaken using a first order polynomial with a nearest neighbor interpolation, incorporating ground control points taken from the GLCF 2005 orthorectified images, producing a Root Mean Square Error (RMSE) of less than 0.5. Ponta Porã site requested

the mosaicing of two images, both were taken in the same dates (August 18,2010). The pre-processing was undertaken with the use of the software ENVI, version 4.7.

2.3. Image classification

The object-based classification (OB) was undertaken using eCognition Developer version 7 software. Six bands of Landsat images were used, excluding thermal infra-red band. A total of seven land cover classes area were identified: water, forest, secondary forest, riparian forest, agriculture, marshes and pasture.

OB classification involved two sub-processes: (i) two segmentations, the first (L1) at the pixel level, and the second (L2) at the object level and (ii) classification.

The selection of segmentation scale parameters is often dependent on subjective trial-and-error methods (Meinel and Neubert 2004). After testing many possible parameters, we used the following: scale: 15, shape: 0.1 and compactness: 0.5. The resulting objects closely corresponded to the boundaries of fields, forests, riparian forests, water bodies and other elements of interest in the image.

After the first segmentation (L1), a hierarchy classification was conducted and several parameters were used: normalized difference vegetation index (NDVI), normalized difference water index (NDWI), shape index and mean difference between bands. The second segmentation (L2) took place in the object level and the parameters used were: scale (100), shape (0.1) and compactness (0.5).

Classification validation was done through a confusion matrix with 30 ground truth control regions of interest by class and by study area to validate OB classification. The confusion matrix was done with focus on pasture class which is our interest class.

2.4. Model for cattle prediction

Database provided from Brazilian Ministry of Agriculture, Livestock and Supply (MAPA) containing the number of farms, total animals per farm and pasture area declared by producers was used to build the prediction model for cattle density. The data from the three municipalities studied were used: Mundo Novo (n=591), Ponta Porã (n=3540) and Porto Murtinho (n=704). Data was treated in the opensource statistical package “R”.

The principle of linear regression is to establish a model linking a variable to be explained, in this study, the “number of cattle” (Nbcattle), by the explanatory variables, here, a single explanatory variable “pasture area” (Apasture). The variables were transformed in order to stabilize variance. It was assumed that the logarithm of response follows a normal distribution; by this way it was possible to construct a classic linear regression model. The second step was to verify the possibility of a linear relationship between Nbcattle and Apasture in each municipality. Farms with no cattle and/or no pasture in the database were excluded.

Comparison between predict value and observed value was made. The validation of the model was done in another data base from three other municipalities in the same region, which are: Caracol, Eldorado and Japorã.

2.5. Mapping cattle density

After the adjustment and validation of the model, the equation was applied to predict animal density in pasture areas detected by remote sensing using OB image classification. Data was treated in ArcGIS software version 10 and maps of cattle density were created. Validation was done through the comparison between cattle density obtained by remote sensing and data base provided by MAPA for each studied area.

3. Results

3.1. Land cover classification

Pasture area counts for 53% of Mundo Novo region, 28% of Ponta Porã and 49% of Porto Murtinho region. By the land cover classification it is possible to note the differences between three municipalities. In Mundo Novo and Porto Murtinho, the main activity is cattle production, in Ponta Porã, principal activity is agriculture, where 48% of the territory is occupied by agriculture areas and bare soil. Total area of Porto Murtinho is smaller than stated by IBGE this is because we used only one Landsat image for Porto Murtinho region, because of the size of municipality we would need three images and it was not the purpose of this study. The biggest part that was not comprised in this study represents the Indian area of 500 thousand hectares (Figure 2).

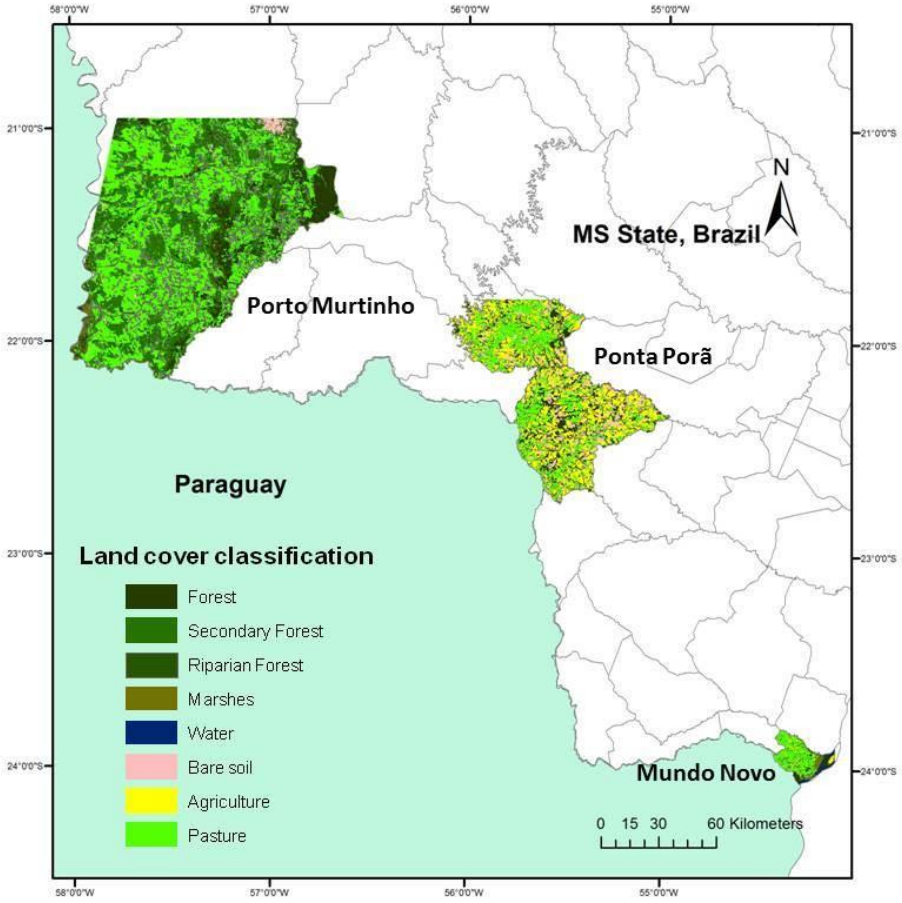


Figure 2. Geographic representation of the subsets of the three municipalities and land cover classification.

In this study we considered that the object-based classification had a good accuracy for the class that we were interested (pasture class) for the three sites studied (Table 1).

Table 1. Producers Accuracy (PA), Users Accuracy (UA) from OB classification from Mundo Novo, Ponta Porã and Porto Murtinho for pasture class and overall accuracy (Acc) and overall Kappa coefficient (KIA)

Pasture class	***Object- based classification***			
	PA %	UA%	Acc	KIA
Mundo Novo	99,8	79,9	75,1	0,69
Ponta Porã	84,2	89,9	74,4	0,66
Porto Murtinho	94,6	87,6	72,6	0,66

MAPA database containing number of cattle producers, total declared area and pasture declared area was confronted with pasture area detected by remote sensing (Table 2). We can observe that magnitude orders are respected for pasture areas even if

there is a small difference between them. For Ponta Porã, pasture area detected by remote sensing was underestimated in 11%, for Mundo Novo and Porto Murtinho were overestimated in 14%.

Table 2. Number of cattle producers, total declared area (ha), pasture declared area (ha) and pasture area detected by remote sensing in Mundo Novo, Ponta Porã and Porto Murtinho

Municipality	Number of producers	Declared total area (ha)	Declared pasture area (ha)	Pasture area detected by RS (ha)	Difference between declared and detected (%)
Mundo Novo	591	25.476	21.391	24.459	14%
Ponta Porã	3.540	259.332	165.292	147.457	-11%
Porto Murtinho	704	738.871	521.177	595.683	14%

3.2. Linear model for predicting cattle density

Porto Murtinho has the biggest pasture area and biggest cattle herd which largely differs from the two other ones (Table 3). Ponta Porã has a great number of familiar settlements (almost 3.000) what makes the average of farms and cattle herd decrease. It was important to select these municipalities that have differences in herd size and farm size in order to have a great heterogeneity for modeling cattle density.

Table 3. Summary of the official database from the three municipalities studied

Municipality	N. of producers*	Declared Pasture area (ha)	Pasture area per farm (ha)	N. of cattle	N. of cattle per farm	N. of cattle per ha
Mundo Novo	452	21.391	36.2	30.299	67	1.4
Ponta Porã	2831	165.292	46.7	230.754	81.5	1.4
Porto Murtinho	585	521.177	740	650.130	1,111	1.2

*Number of producers that had cattle in the year of 2010 and which were used to build the model.

The relationship between the two variables studied (Nbcattle and Apasture) was tested. Figure 3 shows that there is a linear relationship between these two variables. For Ponta Porã there is a concentration of the values between 0 and 4. For Mundo Novo, the values are between 0 and 6 and for Porto Murtinho the values are distributed between 0 and 10. This is because average size of farms is different between municipalities.

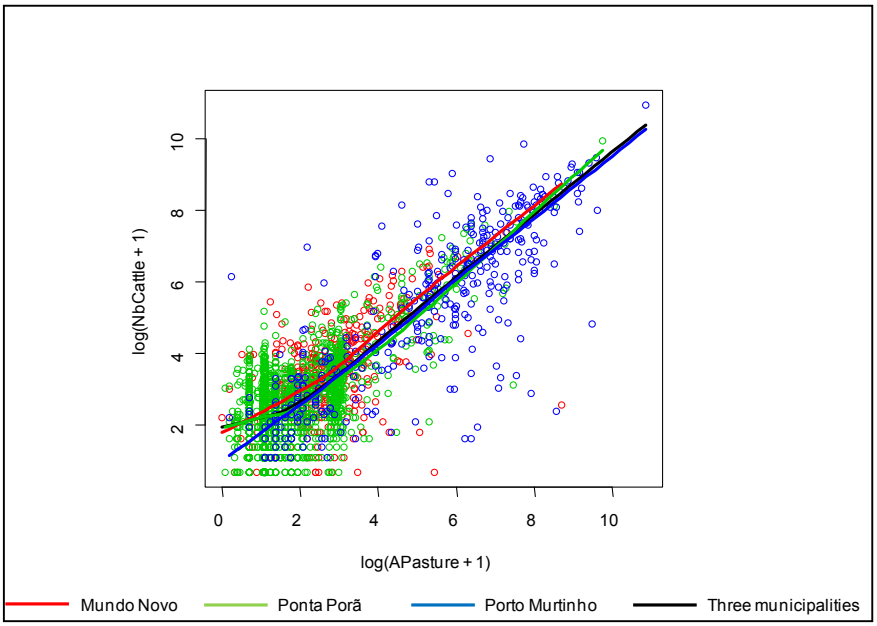


Figure 3. Relationship between pasture area (Apasture) and number of cattle (Nbcattle) in the three municipalities studied (in logarithm).

As the number of cattle (Nbcattle) is related to the pasture area (Apasture), the model built which analyses this relationship is:

log(Nbcattle + 1) ~ log(APasture + 1) + I(log(APasture + 1)*2)
where:
Nbcattle = number of cattle predict
Apasture= pasture area (ha)
I = intercept

Determination coefficient (R^2) was 0.6683, it means that the model was able to predict 67% of the variability in the data set. Porto Murtinho had the biggest difference between observed and predicted (3%) but it can be considered a small difference.

Model validation was done with data from three other municipalities in the same region. There was no difference between predict and observed values (P= 0.9122). As there was no significant difference between observed and predicted for the municipalities used for validation, the model could be applied in other municipalities that exist in the region.

3.3. Applying the model in areas detected by remote sensing

After validation, the model was applied in pasture areas detected by remote sensing (OB) to calculate number of cattle existent in each polygon of pasture for each data subset (Figure 4).

The equation applied to areas detected by remote sensing, and the confident intervals were:

Nbcattle predict =

1.635912 + 0.495649*log (Apastureremot + 1) + 0.031114*log (APastureremot + 1) ^ 2

Conf. Interval min= Nbcattle predicted - 1.96*0.9162738

Confidence interval max = Nbcattle predicted + 1.96*0.9162738

Where:

Apastureremot = area of pasture polygons detected by OB classification

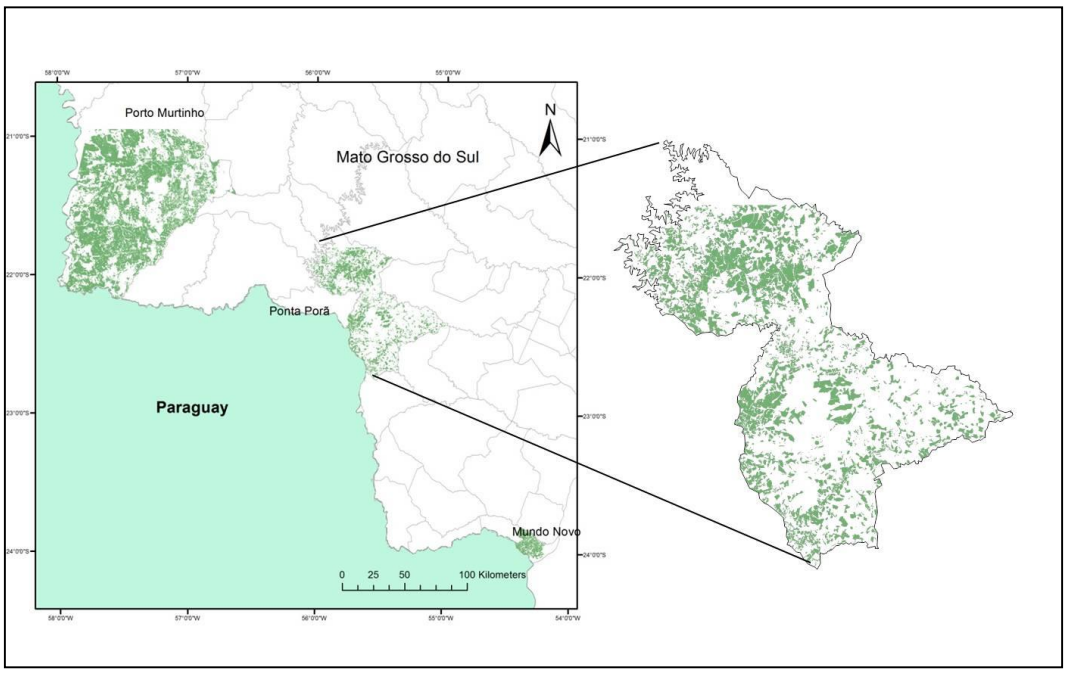


Figure 4. Pasture areas detected by OB classification with Ponta Porã site highlighted

Polygons from pasture area were regrouped in a second segmentation (L2), in order to have bigger areas of similar radiometry, and these areas could represent farms as the cadastre does not exist. Average L2 polygon area was bigger for Porto Murtinho (608 ha), followed by Mundo Novo (436 ha) and Ponta Porã (221 ha). As agriculture is the main activity in Ponta Porã region (48% of land cover) pasture areas are more discontinuous and intercalated with agriculture. Farms usually have both activities, this is the reason why L2 polygons are smaller in Ponta Porã region. Another great difference between municipalities is the number of farms per polygon. In Mundo Novo region there are 10,5 and in the opposite side we have Porto Murtinho with 0.71 farms per polygon. This indicator reflects the characteristic of the municipalities, Mundo Novo with small subsistence producers and Porto Murtinho with extensive cattle areas. Mean pasture size was also bigger for Porto Murtinho (28,5 ha), followed by Ponta Porã with 24,7 ha and Mundo Novo with 23,6. This indicator also reflects the difference between three municipalities in terms of land cover fragmentation.

Table 4 shows the application of the model in areas detected by remote sensing. For Mundo Novo site predicted value was 10% less than the observed value. Great differences in cattle density were seen in Ponta Porã region, 56% more predicted than observed and for Porto Murtinho site the difference between observed and predicted was 23%.

Table 4. Number of cattle observed and predicted by linear model applied in pasture areas detected by remote sensing

Municipality	Observed	Predicted	Min.	Max.
Mundo Novo	29,180	26,869	18,701	36,287
Ponta Porã	105,081	164,970	147,631	185,336
Porto Murtinho	490,370	643,869	587,544	704,512
Total	624,631	835,709	778,631	902,172

Figure 5 represent the map of cattle density (number of cattle per hectare) predicted by the model and applied in pasture areas detected by remote sensing. Average cattle per hectare for Mundo Novo, Ponta Porã and Porto Murtinho according to the prediction model was 1,32; 1,34; and 0.98 respectively, which does not differ from the indicator calculated based on number of cattle and pasture area declared by farmers (Table 2).

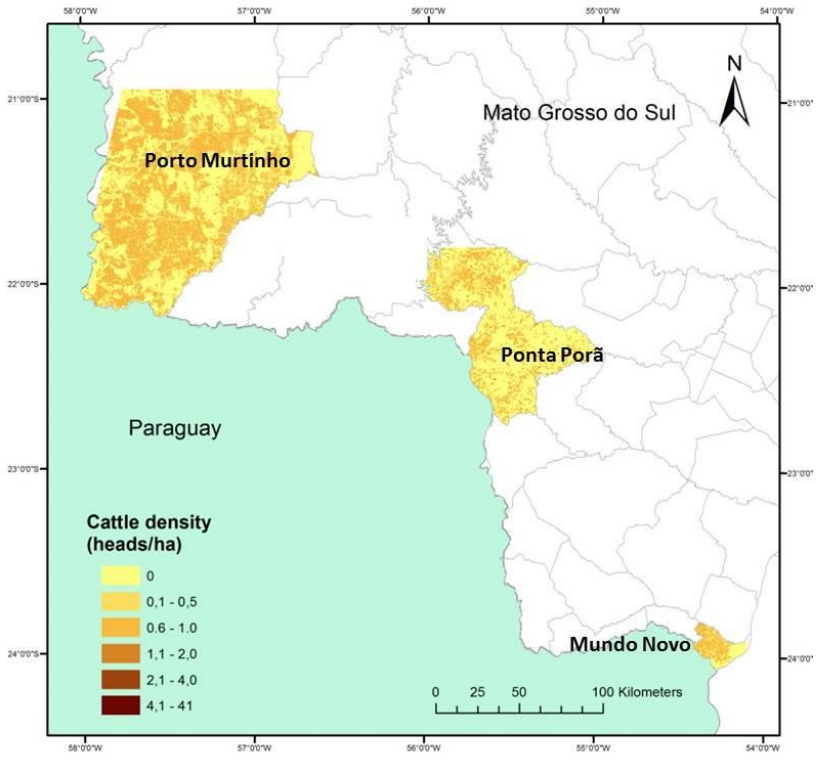


Figure 5. Map of animal density in the three studied sites

4. Discussion

The methodology proposed in this paper, to predict cattle density and location from medium resolution satellite images using object-based classification demonstrated to have good potential use.

OB classification showed good user's and producer's accuracy, as we were interested in extracting pasture areas, OB methodology fitted our purpose. Jobin et al., (2008) noted that one of the advantages of OB is the utility of a knowledge base that is beyond purely spectral information and includes object-related features such as shape, texture and context/relationship, along with the capability to include ancillary data. Myint et al. (2011), found that including principal component images and NDVI within an object-based rule-set classification produced significantly higher accuracy than maximum likelihood classification. The greater producer's and user' s accuracy achieved for pasture class was probably due to the fact of using NDVI and NDWI as one of the indices in the hierarchy classification.

Differences between detected pasture areas by remote sensing and declared areas were observed (Table 2). In Ponta Porã region they were underestimated in 11%. This could have happened due to the common practice in the region that is rotation of cultures. Several farmers have both agriculture and livestock production and part of pasture areas are used in agriculture for one or two years in order to improve soil fertility and after this period they return with cultivated pasture. The overestimation for Porto Murtinho and Mundo Novo sites pasture area was expected. This occurred due to the user's accuracy, although it was high for all three municipalities, we could say that there was 20% in Mundo Novo and 12% of error of inclusion, where other classes such as agriculture or marshes for example were included in pasture class and they do not belong to it. Another factor that could influence these differences is also the database from MAPA. Farm and pasture area are declared by the farmers, Brazilian government has no georeferenced cadastre so declared areas could also be not accurate.

The linear model to predict cattle density (R^2=0.66) can be considered a good predictor because there are other factors associated to cattle density that are not possible to foresee such as economic situation of the farmers and also his decision of having more or less cattle in the farm due to market fluctuations and pasture quality. These factors would count for the others 33% of data variability.

Great differences between observed and predicted cattle density by remote sensing were seen in Ponta Porã region, 56% more predicted than observed. As the model was built based on declared pasture area and cattle number when it is applied to areas detected by remote sensing, it recognizes smaller areas as more populated. As Ponta Porã region has a discontinuous pasture area, and a smaller area of L2 polygons, the model overestimates cattle density.

Another factor associated with this difference is the fact that the model was built in logarithm, and when it is transformed to exponential, the errors are multiplied and when it is estimated the total cattle for each municipality the effect is a sum of exponential errors making the total number superior. For Mundo Novo this effect did

not appeared because total area of the municipality and the number of polygons were much smaller than for the other two municipalities. We can say that the model is good to predict cattle density inside the polygons, but not as good to count for the whole area, as the values are added, errors are added too.

Brazilian government is improving farm and cattle register by implementing farm georeference. The cadastre of all farms in Brazil is expected to be concluded by the year 2025. Even in this region which is considered a model for Brazil in terms of register, lot of work is still needed. In Ponta Porã, for example, from the 3540 farms 620 did not have GPS point location. In Porto Murtinho, from 704 farms, 187 did not have GPS location. In other regions of Brazil, this information does not exist.

As the density maps produced here have better spatial resolution then maps produced by FAO, they would have better usage for supporting public politics related to rural planning and a local development and also epidemiologic studies. The methodology developed here also has potential use for areas where no information is available.

5. Conclusions

Object based classification showed to be an interesting tool to detect and classify pasture areas in the tropics. The results of the study have a great potential to be used in areas where scarce information is available. Limitation of the use is that the model is restricted to the study area. More studies should be done in order to extrapolate the methodology to other regions or other countries which have the same characteristics of cattle grazing or production systems.

6. References

FAO. Food and Agriculture Organization of the United Nations. Global Cattle Density. (2005). Retrieved on 03 june, 2011 from: http://data.fao.org/map?entryId=f8e6a720-88fd-11da-a88f-000d939bc5d8

Ibge. Instituto Brasileiro de Geografia e Estatística (2010).*Cidades: Mundo Novo, MS*. Retrieved on 5 january, 2010 from IBGE/Cidades website: <http://www.ibge.gov.br/cidadesat/link.php?codmun=500568>.

Jobin, B., Labrecque, S., Grenier, M., Falardeau, G. (2008). Object-based classification as an alternative approach to the traditional pixel-based classification to identify potential habitat of the Grasshopper Sparrow. *Environmental Management 41*, 20-31.

Meinel, G., and M. Neubert, 2004. A comparison of segmentation programs for high resolution remote sensing data, International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, XXXV(B4):1097–1102.

Myint, S. W., Gober, P., Brazel, A., Grossman-Clarke, S., Weng, Q. (2011). Per-pixel vs. object-based classification of urban land cover extraction using high spatial resolution imagery. *Remote Sensing of Environment ,115*, 1145-1161.

R Core Team. R: AS Language and Environment for Statistical Computing. R Foundation for Statistical Computing. Vienna, Austria, 2012. Available in: http://R-project.org

Rebanho bovino e bubalino do Brasil, 2010. Ministério da Agricultura e Abastecimento, 2010 Retrieved 09 september, 2010 from the site: http://www.agricultura.gov.br/arq_editor/file/Aniamal/programa%20nacional%20sanidade%20aftosa/Rebanho%20nacional_2010.pdf .

Enhancements to the Bayesian Network for Raster Data (BayNeRD)

Alexsandro C. de O. Silva[1], Marcio P. Mello[2], Leila M. G. Fonseca[1]

[1]National Institute for Space Research (INPE)
Image Processing Division (DPI)
Avenida dos Astronautas 1758 – 12201-010 – São José dos Campos, SP – Brazil

[2]The Boeing Company
Boeing Research & Technology – Brazil (BR&TB)
Estrada Dr Altino Bondesan 500 – 12247-016 – São José dos Campos, SP – Brazil

{acos,leila}@dpi.inpe.br, marcio.p.mello@boeing.com

Abstract. *Bayesian Networks are powerful probabilistic method to make inferences based on evidences. However, this technique has been rarely applied to processing Remote Sensing data. The Bayesian Network for Raster Data – BayNeRD, implemented in the R software, was developed to be used for raster data analysis. This paper describes an enhanced version of the BayNeRD algorithm, stating what has been changed in terms of data pre-processing, user interaction, and outputs.*

1. Introduction

A system that can work with information absence (uncertainty) should be able to assign reliability levels to the variables in its knowledge base and set relations among them. Bayesian Networks (BNs) are probabilistic methods that use graph theory to set relations between variables and probability theory to assign belief levels [Marques and Dutra 2008; Qin et al. 2006].

According to Pearl and Russel (2000), the ability for bidirectional inference, combined with a rigorous probabilistic base, led to the emergence of BNs as a method for reasoning in artificial intelligence and expert system. Indeed, BNs have been used with focus on several applications, such as: medical diagnosis [Kahn et al. 1997; Oniśko and Druzdzel 2013]; speech and gesture recognition [Daoudi et al. 2003; Suk et al. 2010]; and Environmental Science [Castelletti and Soncini-Sessa 2007; Dlamini 2010; Gambelli and Bruschi 2010]. However, even though BN applications have been increasing over the last two decades, Aguilera et al., (2011) and Qin et al. (2006) pointed out that BNs has been rarely used in Earth Observation Science and to process Remote Sensing data.

In this context, Mello et al. (2013) developed a computer-aided Bayesian Network method able to incorporate expert knowledge for the benefit of remote sensing applications and other raster data analysis: Bayesian Network for Raster Data (BayNeRD). It was implemented in R software [R Core Team 2014] and a case study of soybean mapping in Mato Grosso State, Brazil, was used to evaluate its capability to model complex phenomena through plausible reasoning based on data observation. Although the main concepts of BNs for remote sensing applications were implemented by Mello et al. (2013), there are many improvements that can be implemented in order

to create an enhanced version of the BayNeRD algorithm. Thus, this paper describes the main enhancements implemented in the BayNeRD algorithm to improve it as well as the changes in terms of data pre-processing, user interaction, and program outputs.

2. Bayesian Networks and Inference Background

BNs, sometimes called Bayesian Belief Networks [Uusitalo 2007], are mathematical models based on two components: (i) a Directed Acyclic Graph (DAG), and (ii) conditional probability tables (CPTs) [Landuyt et al. 2013].

In a DAG, each node represents a variable in the model, while an arrow linking two variables indicates dependence between them. An arrow starts in a parent variable and ends in a descendent one. As the graph is acyclic, there is no feedback arrows from a descendent to a parent [Landuyt et al. 2013]. One advantage of BNs over other types of predictive models, such as decision trees and neural networks, is that unlike those "black box" approaches, the DAG's arrows represent real connections, not flow of information [Pearl & Russell 2000; Qin et al. 2006]. Figure 1 shows an example of a BN, in which the variable X_1 influences X_2 and X_3. Furthermore, X_3 is influenced by X_2.

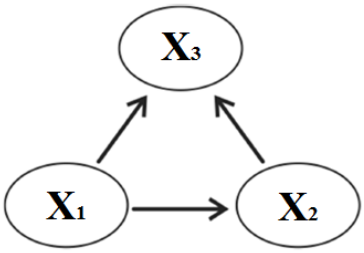


Figure 1. Example of Bayesian Network graphical model.

Once the graphical model is defined, it is necessary to know the probability relations. If a node has no parent, a prior probability function is assigned to it: $P(X_i = x_i)$, in which $x_1, \dots, x_n$ are all possible values (i.e., instantiations) of variable X_i. On the other hand, if a node has parent(s), a conditional probability function is assigned to it. For each variable, the function expresses the probability of that variable being in a particular state (instantiated), given the states of its parents. If x_i denotes the values of the variable X_i and pa_i denotes the set of values for X_i's parents, then $P(x_i|pa_i)$ denotes the conditional probability. For example (see Figure 1), $P(X_3 = x_3|X_1 = x_1, X_2 = x_2)$.

The fundamental rule of conditional probability is

$$P(a|b) = P(a,b)/P(b) \quad (1)$$

or

$$P(a|b) * P(b) = P(a,b) \quad (2)$$

in which $P(a,b)$ is the joint probability of event $a \wedge b$.

It is known that $P(a,b) = P(b,a)$. Thus, from Eq. 2 we have

$$P(a|b) * P(b) = P(b|a) * P(a) \quad (3)$$

that will result in

$$P(a|b) = P(b|a) * P(a)/P(b). \quad (4)$$

Eq. 4 is the Bayes' theorem [Neapolitan 2003], which is the core of a BN, allowing probabilities to be updated under the light of a new evidence. Indeed, the Bayes' theorem updates the knowledge (priori probability) of an event considering new/additional evidence (conditional probability), allowing one to have a plausible reasoning based on a degree of belief (posteriori probability). This ability to compute posteriori probabilities given some evidence is called inference.

The effectiveness of a BN lies in the possibility to compute through Bayes' theorem not only the probability distributions for descendent variables given the values of their parents, but also the distributions of the parents given the values of their descendants. That is, the BN allows to know the effects given the causes and the causes given the effects [Uusitalo 2007; Aguilera et al. 2011].

The joint distribution is computed by the product of prior and conditional probabilities for each variable given its parents, as

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i|pa_i) \quad (5)$$

in which x_i is the value observed for the variable X_i and pa_i is the set of values for X_i's parents.

3. The original BayNeRD algorithm

The first version of the BayNeRD algorithm implemented by Mello et al. (2013) is available on Internet. Figure 2 shows the procedures used to apply the BayNeRD system in a study case to identify soybean plantations in Mato Grosso State, Brazil.

The BayNeRD algorithm has been implemented in R software [R Core Team 2014]. The algorithm handles raster data in GeoTiff format, where each GeoTiff corresponds to a variable (node) of the network. The variable that represents the phenomenon under study is named *target variable* and the remaining variables are named *context variables*.

There are packages already implemented in R software to handle spatial data. GeoTiff files were originally loaded by *rgdal package* [Bivand et al. 2014], which makes bindings to Geospatial Data Abstraction Library (GDAL).

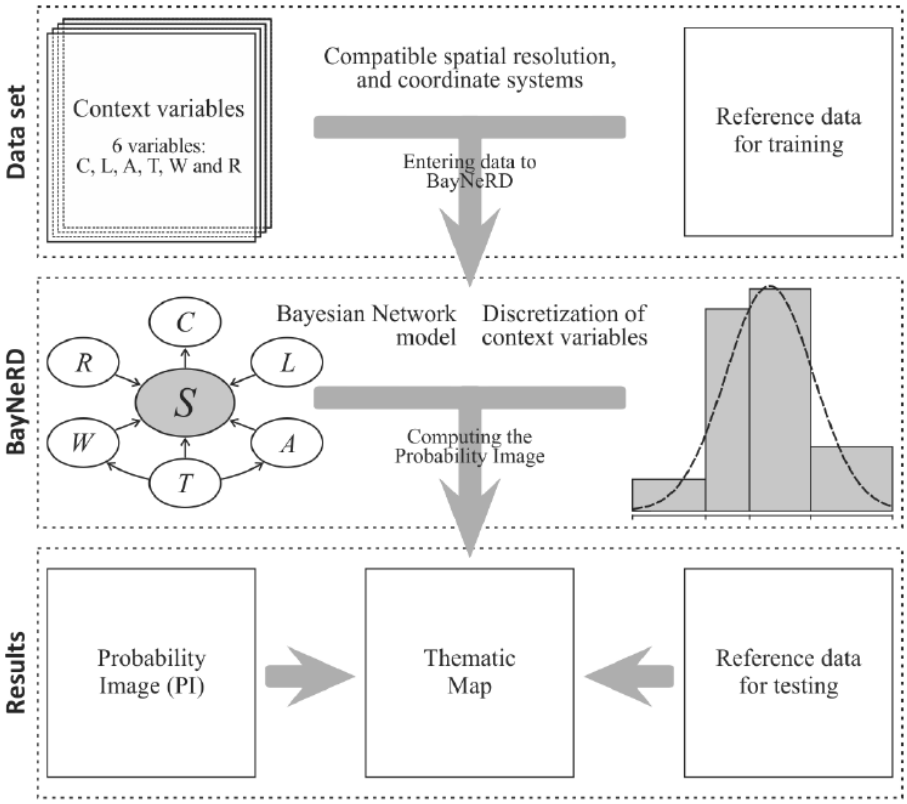


Figure 2. Procedures for using BayNeRD in remote sensing applications [Mello et al. 2013].

3.1. Target and Context Variables

The GeoTiff that represents the target variable must be provided as reference data for training. Its pixels can be categorized in four labels: (i) target presence; (ii) target absence; (iii) missing data (no observation); and (iv) pixels outside the study area. Although the target variable GeoTiff may contain more than four labels, it must have at least two: (i) and (ii) [Mello et al. 2013].

The context variables can be either numerical (e.g., pixels with digital numbers) or categorical (e.g., thematic map); and they can contain also missing data. The context variables can have dependence relationship with target variable and/or among themselves. Furthermore, in the original version of BayNeRD, all GeoTiffs (for the target variable as well as for all context ones) must be compatible in relation to pixel size and coordinate system and all variables (target and context) must represent the same geographic area.

3.2. Bayesian Network Graphical model

After reading the target and the context variables into BayNeRD, the algorithm interacts with the user to design a BN graphical model. In the original version, the user is asked about the dependence relations among all variables. As dependencies are represented by arrows in a DAG, BayNeRD asked if there was an arc from a variable to all the others (pairwise).

Among the packages already implemented in R software to compute Bayesian analysis, the *catnet package* [Balov and Salzman 2013] has been developed to handle discrete BNs.

3.3. Discretization and Probability Functions

One of the main difficulties of applying BNs for real problems is to define the probability functions. According to Mello et al. (2013), BayNeRD was developed to interact with the user to define the probability functions through discretization processes of the context variables based on his/her knowledge. In the discretization, observed values of a variable are represented by discrete quantities (similar to process of drawing a histogram). In other words, the range of observed values is divided into intervals defined by the user.

In BayNeRD, the number of intervals can be computed following three criteria: (i) equidistant, in which intervals with the same width, (ii) quantile, in which intervals tends to have the same number of elements (pixels), and (iii) manually, in which the user defines the limits of each interval. The discretization will impact the probability functions, which are computed through pixel counting to both the dependence relations defined in the graphical model and the intervals defined in the discretization process. The user should be skilled to define the suitable intervals for each context variable so that all scenarios (i.e., combination of variable's and its parent's intervals) have representative data to compute the probability functions [Mello et al. 2010]. Besides, defined probability functions shall be representative of the real probability functions.

3.4.Probability Image

When the probability of target occurrence is computed for each pixel inside the study area, given a specific scenario, a Probability Image (PI) is formed. The PI consists of a raster data and it is the main result of BayNeRD.

Suppose that in the example of Figure 1 the variable X_3 is the target variable. The probability image will be formed by computing

$$P(X_3 = 1 | X_1 = x_1, X_2 = x_2) \quad (7)$$

for each pixel in the study area; where 1 is used to represent target presence.

4. BayNeRD enhanced version

To enhance the BayNeRD algorithm and to keep it updated with new R packages, several improvements have been added to it. Following, we will describe the changes implemented in BayNeRD through a comparative process between the original and the enhanced version.

As mentioned before, there are some packages that handle raster data in the R software [R Core Team 2014] (such as *rgdal package*, used in the BayNeRD original version). The *'raster' package* [Hijmans et al. 2014] provides several high-level raster data manipulation functions and showed to be more suitable for BayNeRD. A noteworthy feature of this package is that it allows working better with large raster datasets stored on disk, which makes the processing faster, since the package does not read all the values of cell's raster into memory. The package creates objects from large

files that only contain information about the data (e.g., filename, pixel size, extent) and, in computations with the pixels, the data are processed in chunks [Hijmans 2014].

Considering the advantages of *'raster' package* over the *rgdal package*, we added it into BayNeRD in order to optimize the raster data (GeoTiff) processing for both target and context variables.

4.1. Target variable

In the original BayNeRD version, the target variable is instantiated in two values: 1 and 0, representing presence and absence, respectively. In the enhanced BayNeRD version, the target variable may have more than two instantiations. For instance, the target variable raster may have label 1 for the thematic class A, label 2 for the thematic class B and 0 for the target absence. This change will affect the definition of probability functions and BayNeRD's outputs. In addition, the raster data may still contain labels representing missing data and also labels representing pixels outside the study area as before, which will be used only to mask out pixels outside the study area in the context variables.

4.2. Context variables

One of the main benefits of enhancing BayNeRD with the *'raster' package* is to reduce the preprocessing data step. In the enhanced BayNeRD version, raster data may have different structures unlike the original version, in which all raster data must have compatible pixel size, extent (number of rows and columns) and coordinate reference system.

Each input context variable in the BayNeRD system is transformed to be compatible with target variable. That is, through *'raster' package* functions the algorithm is able to transform the coordinate reference system, to resample the spatial resolution, and to intersect the raster data according to the bounding box of target variable as shown in Figure 3. In this case, a new raster data is created following the target variable's bounding box. The pixels are interpolated from the context variable using the nearest neighbor method to minimize distortions. If a region has no intersection between the context and the target variables, their pixels are filled with NA values (Not Available), indicating missing value.

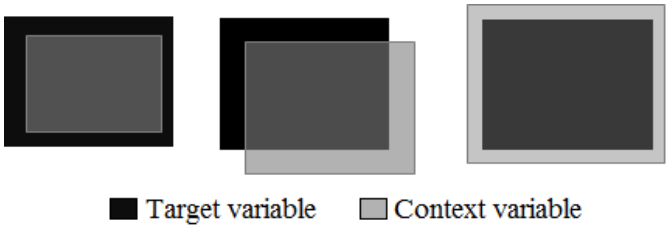


Figure 3. Different possibilities for intersection between context variable and target variable.

4.3. Bayesian Network Graphical model

The interactive process to state dependence relationships among the variables was substantially enhanced in BayNeRD. Before improvements, R interacted with the user by command line asking whether an arc exist for each pairwise variable. That is, for

each context variable the algorithm ask $(n-1)$ questions, in which n is the number of variables. Therefore, the user must answer $n*(n-1)$ questions, that is, the growth of configuration questions is exponential.

In the enhanced BayNeRD, we implemented a more efficient way to define dependence relationships among the variables. The R software has also packages to handle with DAGs, and to perform Bayesian analysis. We created a new BN graphical model interface through the integration of both packages *'deal'* [Bottcher and Dethlefsen 2013] and *'bnlearn'* [Scutari 2009]. The first one allows the user to specify a BN through a point and click interface and the second one is used for Bayesian analysis and inference. Figure 5 shows the new interface implemented in the enhanced BayNeRD version.

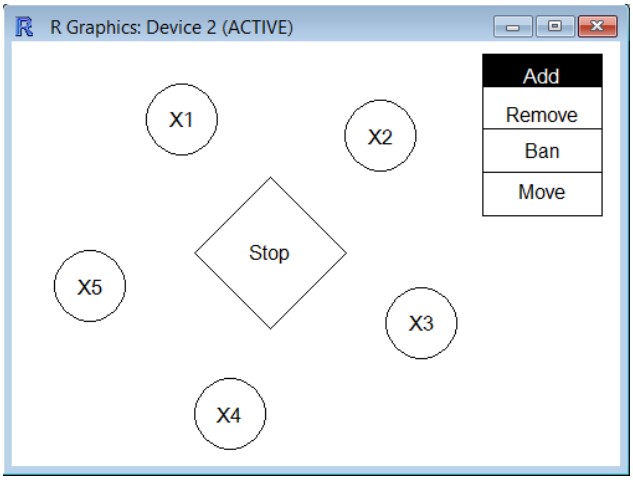


Figure 4. Interface to design the Bayesian Network graphical model.

In this interface, the user can insert or remove arcs between pairwise variables using either 'Add' or 'Remove' buttons and clicking on the variables. To insert/remove an arrow from node X_1 to node X_2, first click on "Add"/"Remove" button, then click on node X_1 and then click on node X_2. The interface also helps the user to avoid mistakes. For example, if the user enters an arrow to create a cycle it will not be drawn since the BN graphical model shall represent a DAG (acyclic). Therefore, the task of BN graphical model building becomes now easier and quicker compared to the previous BayNeRD version.

4.4. Discretization and Probability Functions

During the discretization process, the intervals can be defined by the user or following the criteria mentioned earlier – equidistance and quantile. These criteria were kept in the enhanced BayNeRD version and additional criterion was added. Now, the user may choose clustering criteria to convert continuous variables into categorical variables. This criteria is an unsupervised method through k-means clustering to partition the observed values of each context variables into k clusters defined by user, in which each value belongs to the cluster with the nearest mean [Hahsler et al. 2014].

The range of observed values for a context variable are divided into intervals, defined by lower and upper limits, and each interval is encoded as one category. Thus, a context variable will have n categories if it is discretized into n intervals. The

probability functions are computed by counting categorical pixels for each context variable and taking into account their (in)dependence relationships. However, the user must also perform the discretization process for the target variable once the target variable may contain more than two labels in the enhanced BayNeRD version.

4.5. Probability Image

The number of instantiated values for the target variable defines also the number of output layers in the Probability Image (PI). Let's suppose that the variable X_3 (in Figure 1) is the target variable and the phenomenon occurrence is represented by three thematic classes: $X_3 = 1$, $X_3 = 2$ or $X_3 = 3$. In this case, we have one output layer with probability values for each class (presence). Each output layer is now called Probability Band (PB), and the stacking of PBs creates the Probability Image (PI) in the enhanced BayNeRD version.

Through the *bnlearn package*, queries are performed into CPTs and the system returns the probability of an event, given all observed evidences. A combination of evidences stated by the instantiation of all observed context variable is called scenario. Given the example in Figure 1, the PBs are computed, respectively, as

$$P(X_3 = 1|X_1 = x_1, X_2 = x_2) \quad (8)$$

$$P(X_3 = 2|X_1 = x_1, X_2 = x_2) \quad (9)$$

$$P(X_3 = 3|X_1 = x_1, X_2 = x_2). \quad (10)$$

For a given scenario, there are three events to compute associated probabilities. Consequently, the PBs are computed simultaneously to optimize the algorithm. If any context variable has missing data for any specific pixel, the probability for this pixel is computed anyway. It is even possible to compute $P(X_3 = x_3)$ for pixels without any observation through the priori probability, as in the original version.

5. Conclusion

The Bayesian Network for Raster Data – BayNeRD proposed by [Mello et al., 2013] represents a new probabilistic approach for raster data applications. This paper described the improvements made upon the original BayNerD algorithm that provided an enhanced BayNeRD version. Although this version has not been extensively tested, we do believe that it is more efficient and more user-friendly.

BayNeRD was enhanced to best handle raster data, also minimizing preprocessing of raster images. Indeed, the enhanced BayNeRD version handles raster datasets with different coordinate systems, spatial resolution and extent. A new BN graphical model interface is the most noticeable improvement in BayNeRD. It was also added the ability to model more than two possible instantiations for the target variable. The algorithm computes the probability for each possible target value given the observations made upon the context variables and then creates several Probability Bands (PB) that, when stacked, produce a Probability Image (PI) as outcome.

As future research, we plan to include spatial features (neighborhood information, for example) in the probability computation, and also to improve the algorithm efficiency by using parallel processing techniques. Also, we plan to exhaustively test the enhanced version comparing to the original one. The enhanced

BayNeRD version is still under construction and it will be available on Internet as soon as it is ready.

Acknowledgements

The authors thank CNPq (134400/2013-5) for financial support to first author; and the reviewers for their valuable comments and inputs.

References

Aguilera, P. a., Fernández, a., Fernández, R., Rumí, R., & Salmerón, a. (2011). Bayesian networks in environmental modelling. *Environmental Modelling & Software*, *26*(12), 1376–1388. doi:10.1016/j.envsoft.2011.06.004

Balov, N., & Salzman, P. (2013). catnet: Categorical Bayesian Network Inference. Retrieved from http://cran.r-project.org/package=catnet

Bivand, R., Keitt, T., & Rowlingson, B. (2014). rgdal: Bindings for the Geospatial Data Abstraction Library. Retrieved from http://cran.r-project.org/package=rgdal

Bottcher, S. G., & Dethlefsen., C. (2013). deal: Learning Bayesian Networks with Mixed Variables. Retrieved from http://cran.r-project.org/package=deal

Castelletti, a., & Soncini-Sessa, R. (2007). Bayesian Networks and participatory modelling in water resource management. *Environmental Modelling & Software*, *22*(8), 1075–1088. doi:10.1016/j.envsoft.2006.06.003

Daoudi, K., Fohr, D., & Antoine, C. (2003). Dynamic Bayesian networks for multi-band automatic speech recognition. *Computer Speech & Language*, *17*(2-3), 263–285. doi:10.1016/S0885-2308(03)00011-1

Dlamini, W. M. (2010). A Bayesian belief network analysis of factors influencing wildfire occurrence in Swaziland. *Environmental Modelling & Software*, *25*(2), 199–208. doi:10.1016/j.envsoft.2009.08.002

Gambelli, D., & Bruschi, V. (2010). A Bayesian network to predict the probability of organic farms' exit from the sector: A case study from Marche, Italy. *Computers and Electronics in Agriculture*, *71*(1), 22–31. doi:10.1016/j.compag.2009.11.004

Hahsler, M., Buchta, C., Gruen, B., & Hornik, K. (2014). arules: Mining Association Rules and Frequent Item sets. Retrieved from http://cran.r-project.org/package=arules

Hijmans, R. J. (2014). Introduction to the "raster" package (version 2.2-31). Retrieved from http://cran.r-project.org/web/packages/raster/vignettes/Raster.pdf

Hijmans, R. J., Etten, J. van, Mattiuzzi, M., Sumner, M., Greenberg, J. A., Lamigueiro, O. P., … Shortridge, A. (2014). raster: Geographic data analysis and modeling. *R Package Version 2.2-12*. Retrieved from http://cran.r-project.org/web/packages/raster/

Kahn, C. E., Roberts, L. M., Shaffer, K. A., & Haddawy, P. (1997). Construction of a Bayesian network for mammographic diagnosis of breast cancer. *Computers in Biology and Medicine*, *27*(1), 19–29. doi:10.1016/S0010-4825(96)00039-X

Landuyt, D., Broekx, S., D'hondt, R., Engelen, G., Aertsens, J., & Goethals, P. L. M. (2013). A review of Bayesian belief networks in ecosystem service modelling. *Environmental Modelling & Software*, *46*, 1–11. doi:10.1016/j.envsoft.2013.03.011

Marques, R. L., & Dutra, I. (2008). Redes Bayesianas : o que são , para que servem , algoritmos e exemplos de aplicações. Retrieved from http://pt.scribd.com/doc/3837765/Bayesianas

Mello, M. P., Risso, J., Atzberger, C., Aplin, P., Pebesma, E., Vieira, C. A. O., & Rudorff, B. F. T. (2013). Bayesian Networks for Raster Data (BayNeRD): Plausible Reasoning from Observations. *Remote Sensing*, *5*(11), 5999–6025. doi:10.3390/rs5115999

Mello, M. P., Rudorff, B. F. T., Adami, M., Rizzi, R., Aguiar, D. A., Gusso, A., & Fonseca, L. M. G. (2010). A simplified Bayesian Network to map soybean plantations. In *2010 IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2010)* (pp. 351–354). Honolulu, HI, USA: IEEE. doi:10.1109/IGARSS.2010.5651814

Neapolitan, R. E. (2003). *Learning Bayesian Networks* (p. 674). Pearson Prentice Hall.

Oniśko, A., & Druzdzel, M. J. (2013). Impact of precision of Bayesian network parameters on accuracy of medical diagnostic systems. *Artificial Intelligence in Medicine*, *57*(3), 197–206. doi:10.1016/j.artmed.2013.01.004

Pearl, J., & Russell, S. (2000). Bayesian Networks. *Department of Statistics Papers*. Departament of Statistics, UCLA. Retrieved from https://escholarship.org/uc/item/53n4f34m

Qin, D., Jianwen, M., & Yun, O. Y. (2006). Remote sensing data change detection based on the CI test of Bayesian networks. *Computers & Geosciences*, *32*(2), 195–202. doi:10.1016/j.cageo.2005.06.012

R Core Team. (2014). R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing. Retrieved from http://www.r-project.org/

Scutari, M. (2009). Learning Bayesian Networks with the bnlearn R Package. *Journal of Statistical Software*, *35*(3), 22. Machine Learning. Retrieved from http://arxiv.org/abs/0908.3817

Suk, H.-I., Sin, B.-K., & Lee, S.-W. (2010). Hand gesture recognition based on dynamic Bayesian network framework. *Pattern Recognition*, *43*(9), 3059–3072. doi:10.1016/j.patcog.2010.03.016

Uusitalo, L. (2007). Advantages and challenges of Bayesian networks in environmental modelling. *Ecological Modelling*, *203*(3-4), 312–318. doi:10.1016/j.ecolmodel.2006.11.033

Geovisualization of the Academic Trajectories of Brazilian Researchers

Caio Alves Furtado, Thamara Karen Andrade, Clodoveu A. Davis Jr.

Departamento de Ciência da Computação – Universidade Federal de Minas Gerais (UFMG) – Belo Horizonte – MG – Brazil

{caioaf, thamara, clodoveu}@dcc.ufmg.br

Abstract. *People seeking academic careers usually pursue degrees in various institutions, looking for the best combinations regarding their intellectual interests, their personal means and the opportunities that arise in their lives. This can be perceived by looking at the stages in the education of current researchers, as recorded in their curricula vitae. We define the academic trajectories of researchers as the sequence of degrees obtained by a researcher, culminating with a work position. After geolocating academic and work institutions, and using the time period associated with each degree in the vitae data, the trajectories become spatiotemporal objects. We present an analysis of the academic trajectories of nearly 6,000 researchers associated with CNPq's National Institutes of Science and Technology program, based on their curriculum vitae data. An interactive visualization tool has been developed. Analyses include the variety of trajectories, preferred destinations, variations along time and different patterns related to research area.*

1 Introduction

Lattes[1] is a Web-based application created by CNPq (the Brazilian National Council for Scientific and Technological Development) to collect and integrate a wide range of information about the Brazilian academic community. One of its services is *Currículo Lattes*, a curriculum vitae Web system for researchers and students. All information in Lattes is publicly available, and currently covers the vast majority of active Brazilian researchers, groups, and institutions. Lattes is a rich database on Brazilian scientific research, built from the perspective of the individuals involved in it.

CNPq, along with CAPES (the Brazilian Ministry of Education's organization for graduate courses and curricula) and regional foundations, also created a program to foster and promote Brazilian research groups, called *Institutos Nacionais de Ciência e Tecnologia*[2] (National Institutes of Science and Technology, or INCT). The program created 101 institutes, covering thematic areas deemed strategically important for the country. While the INCT program does not cover every research group in the country, it includes many of the best Brazilian research groups.

CiênciaBrasil[3] is one of the research projects conducted within one of the INCTs. The project built a portal that uses Lattes curriculum vitae (CV) data to

[1] Lattes: http://lattes.cnpq.br

[2] INCT: http://estatico.cnpq.br/programas/inct/_apresentacao/apresentacao.html

[3] CiênciaBrasil: http://pbct.inweb.org.br

configure a research-based social network, in which relationships are characterized by collaboration in publications. This allows CiênciaBrasil to study many aspects of the researchers' careers and activities, seeking patterns from practices and behaviors captured from scientific production and collaboration. Currently, the portal includes CVs from INCT researchers that hold a Ph.D., but contents are constantly expanding.

Our work involves studying and characterizing patterns of placement and mobility of Brazilian researchers through time. We characterize events in the researchers' careers as points in space and in time, thus defining *academic trajectories*, comparable to trajectories of moving objects. In both cases trajectories are described by a set of positions collected through a time period, but the average academic trajectory has fewer positions and spans a much larger time interval, covering years. In this paper, we explore alternatives for using geographic visualization techniques to provide tools for exploratory analysis of such academic trajectory data.

This paper is organized as follows. Section 2 discusses some related work. Section 3 describes our dataset and the steps taken in its preparation. Section 4 presents several analyses, with a variety of visualization alternatives, exploring multiple aspects of the problem. Finally, Section 5 presents a final discussion and indicates future work.

2 Related Works

This paper explores the spatiotemporal academic trajectories of INCT researchers, including stages such as graduation, graduate studies, employment in a university or research institution, and aims to discover and discuss the patterns in such trajectories. We focus in understanding the diversity in the researchers' formation, analyzing the trajectory patterns and their changes through time.

CiênciaBrasil is an INCT project that studies Brazilian researchers and research groups (Laender, Moro et al. 2011). It uses Lattes CV data and focuses on each researcher's publications and collaborations, which define a co-authorship network. This work uses CiênciaBrasil data, and explores the researchers' academic historic and their spatio-temporal aspect, also expanding the analysis to institutions.

Some works study the mobility of researchers. Van Bouwel (2010) discusses the international mobility decisions of researchers that obtained a Ph.D. in Economics in the United States. The report verifies the final work destinations of researchers after getting their Ph.D.s and explores the factors that drive their movement, showing that about 50% of them remain in the USA after getting their degrees, while only one third of the remaining half returns to their home country. A report from the European Commission (IDEA Consult 2010) details the mobility patterns of European researchers and discusses the influencing factors and impacting effects of their mobility. Our study focuses a smaller group, most of which employed in Brazil, but with a variety of academic trajectories up to their participation in an INCT.

Schich et al. (2014) developed a study on the mobility of a large number of people through history. Gathering birth and death dates and places from FreeBase biographies and other sources, the authors reconstructed simple two-point trajectories of more than 150,000 notable individuals. Visualization techniques, including animation, were used to show patterns in each period in human history, by identifying sources and destinations of the lives of so many people.

Câmara et al. (2004) consider that the emphasis of spatial data analysis should be on measuring properties and relationships, explicitly considering the spatial location as part of the studied phenomenon. In our analysis, we concentrate on the institutions in which the current INCT researchers studied and worked, and show how the trajectory between them can be influenced by certain aspects of the researcher's CV. Visualization tools are often used as part of spatial data analysis, especially in the exploratory phase. Interactive scientific data visualization, as presented in this work, provides means to discover data properties (Haining 2003).

An academic trajectory is similar to a mobile object trajectory (Etienne, Devogele et al. 2012). The trajectory can be defined as a set of positions that correspond to places where the researcher obtained a degree in his education, associated with a time period. These positions are composed of a timestamp and a spatial coordinate, corresponding to the location of the institution at which the degree was obtained, and the chronological ordering of the positions forms the trajectory. The temporal granularity of an academic is measured in years, and time periods potentially cover several decades. A researcher will rarely have more than five positions in his trajectory.

Spaccapietra et al (Spaccapietra, Parent et al. 2008) discuss the semantics of trajectories in more detail. They describe three types of trajectories: *metaphorical* (steps in an evolutionary process), *naïve geographical* (major geographic points along a path), or *spatiotemporal* (spatial coordinates associated to timestamps). According to this classification, academic trajectories are metaphorical, since stops correspond to stages in a researcher's education, but naïve geographical concepts appear when analyzing the location of the affiliation institutions along the career. Furthermore, Spaccapietra et al. define trajectories as collections of stops and moves. In the case of academic trajectories as defined here, we are mostly interested in the moves.

3 Dataset

Our dataset is composed of a set of researchers' Lattes CVs, or profiles. The profiles are publicly available for download as XML files, and were collected from December 2012 through August 2013 as part of the CiênciaBrasil project. For now, the CiênciaBrasil portal covers only researchers associated with an INCT. Researchers are required to have and to keep up-to-date their vitae in Lattes, as a precondition to applying for grants and other forms of financing. Considering that 94.7% of the profiles were modified between January 2012 and August 2013, we consider the information provided by the profiles is correct and current enough for our studies.

The Lattes profile has academic information on the user, such as his name, workplace, and a list with his graduation degrees. Each degree has an institution associated with it, indicating where said researcher concluded his graduation, and the starting and ending years. We consider the set of institutions attended by a person, along with the time periods in which she attended, to be his *academic trajectory*. The trajectory is chronologically ordered. The expected trajectory is *bachelor* → *master* → *doctorate* → *pos-doctorate*, but this order isn't always verified. Every researcher in the database has at least one Ph.D. degree, but the number of degrees and the order in which he completed them may change. A researcher can have two doctorates and no pos-doctorate, or can have his doctorate course right after his bachelor, and get a master's degree after that, as indicated by the years associated with each degree.

Figure 1 shows a simplified schema for the data, after collection and geocoding. The block corresponding to each degree in the researcher's data is multivalued. Additional data on the researcher can be obtained using the Lattes ID, since the entire contents of the CV are stored elsewhere in CiênciaBrasil. INCTs are classified into 8 research areas, defined by CNPq: agriculture, environment, energy, exact sciences, humanities, nanotechnology, heath/medical, and engineering/information technology (IT). Researchers can be associated with more than one INCT.

The database is composed of 5,973 researchers, all of which associated with at least one INCT. There are 3,478 unique institutions, including academic institutions and employment institutions. From these, 2,858 institutions appear exclusively as places of education, 399 institutions that appear exclusively as employers, and 221 institutions that are both academic and employers.

```
Researcher(LattesID, [degreeType, institutionID,
    startYear, endYear], workInstitutionID)

Institution(institutionID, institutionName, location)

INCT(inctID, inctName, inctArea)

INCTResearchers(inctID, LattesID)
```

Figure 1. Data schema

The distribution of researchers throughout the institutions is uneven. More than 50% of the researchers work in only 15 of the 620 employment institutions. This behavior is also found in academic entries, in a characteristic long-tail distribution. Analyzing individual trajectory segments, there are 21,092 segments for the 5,973 researchers, about 3.5 segments per researcher. Of those, 7,930 segments connect an institution to itself, thus representing a continuation of studies in the same institution.

To materialize the academic trajectory as a spatiotemporal object, the institutions need to be geographically located, but Lattes does not provide the institutions' addresses or coordinates. We geocoded the institutions based on their names, as supplied in Lattes. The institution's name is processed as given by the user, thus errors and typos may occur. In the case of institutions with multiple campuses, Lattes usually does not define a specific one. Therefore, we considered the institution's main campus as a default. We selected a set of institutions to check the geocoding process. We picked every institution with at least 5 researchers associated with it (84 institutions), corresponding to 80.59% (4,878) of the recorded number of PhD degrees, and manually checked their location, correcting it when needed. Only three institutions could not be geocoded, due to faulty information, corresponding to 61 researchers.

4 Analysis

We created an interactive Web map application[4] to show researcher trajectories. We used a Google Maps layer as a background, and a Javascript module to draw the trajectories. There are filters on some dimensions, such as INCT area, type of degree, and trajectory segment final year. Some controls on the appearance of trajectories are also available, such as varying the thickness of lines according to the number of

[4] http://aqui.io/trajectory/

equivalent segments, and hiding segments that connect Brazil and foreign institutions. In the next subsections, we show steps in an exploratory analysis of the described dataset, using geovisualization in an interactive environment.

4.1 Individual trajectory

The trajectory of any given individual can be shown in the interactive map, by supplying the researcher's Lattes ID. Figure 2 shows an example, in which the researcher starter his career at Fortaleza, in the Brazilian northeast, where he received his bachelor's degree. Then he moved to Campinas, to get a master's degree, next to the USA, for a Ph.D. Getting back to Brazil, he had a post-doctorate stage at Belo Horizonte, after which he was hired also in Belo Horizonte as a professor in a university. The last segment is a null link, i.e., a new stage in the researcher's career, but at the same location.

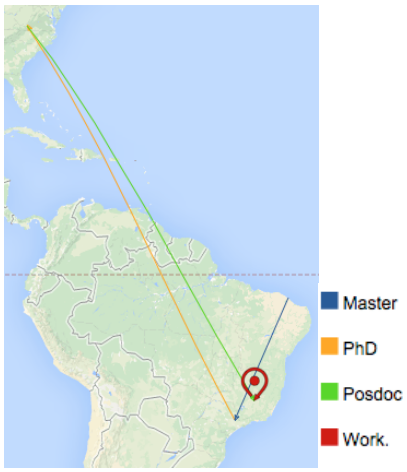


Figure 2. An individual trajectory

4.2 Frequent destinations

Figure 3 shows all trajectories simultaneously. It is apparent that most trajectories link Brazil to the United States or Europe, but most links are null.

Table 1 shows the total number of segments grouped into the most frequent origins and destinations. Naturally, most destinations are within Brazil, but the number of segments that involve destinations in the United States or Europe is expressive, while segments including Africa, Asia or Oceania are infrequent.

Table 1. Start and end points of trajectory segments

Start \ End	Brazil	USA	Europe	Other
Brazil	14,373	1,026	1,282	424
USA	976	385	134	47
Europe	1,244	83	579	50
Other	254	41	57	137

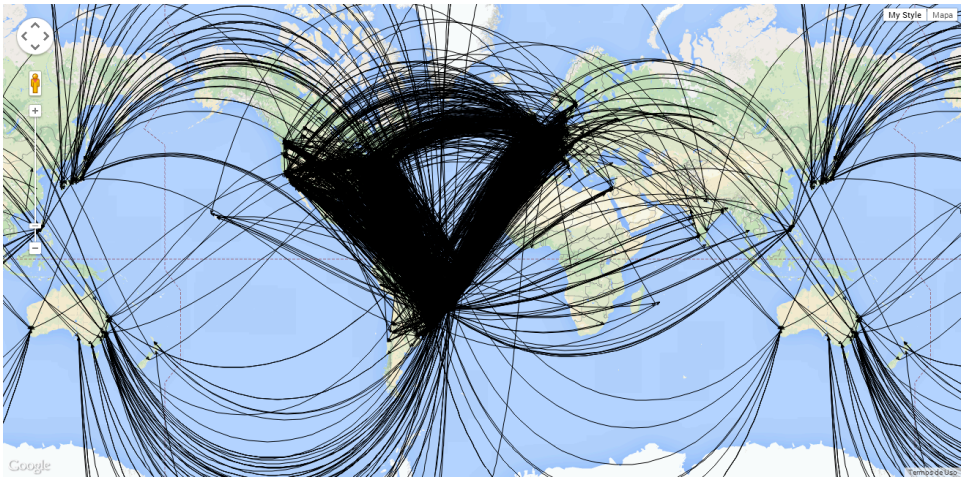
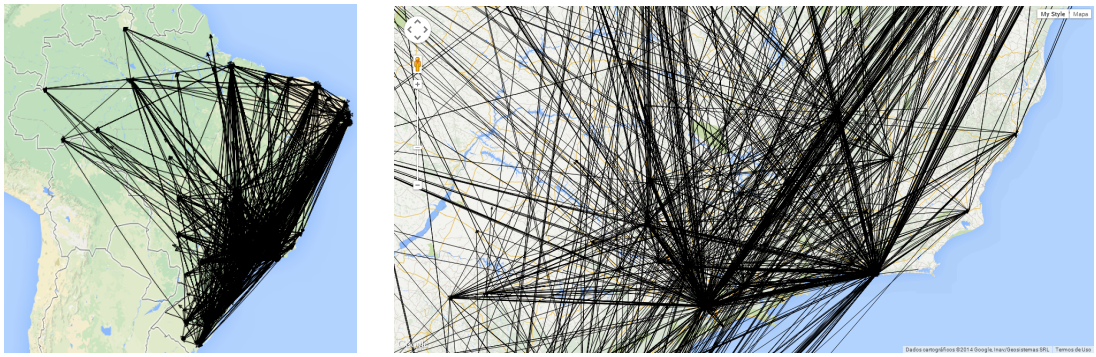


Figure 3. Existing trajectories worldwide

Figure 4 shows trajectory segments with both ends in Brazilian institutions, indicating successive stages in the researcher's education that were carried out in Brazilian institutions. Notice (Figure 4b) the concentration of trajectory segments in southeastern institutions, especially in São Paulo and Rio de Janeiro. Table 2 shows that these states, plus Minas Gerais, Rio Grande do Sul and Pernambuco, concentrate 78% of the destinations within Brazil. Notice also that most segments start and end in the same state. Next section explores this tendency further.

Table 2. Most common start and end points within Brazil

Start \ End	SP	RJ	MG	RS	PE	Others
São Paulo (SP)	4,553	169	163	57	56	620
Rio de Janeiro (RJ)	227	2,070	54	21	13	171
Minas Gerais (MG)	225	102	1,035	17	8	147
Rio Grande do Sul (RS)	144	45	14	883	9	120
Pernambuco (PE)	89	30	10	9	292	69
Others	612	184	98	73	77	1,907



(a) **(b)**

Figure 4. Existing trajectories (a) in Brazil and (b) Southeast region close-up

4.3 Null trajectory segments

Many researchers fulfill consecutive stages in their education at the same institution. As a result, there are many null trajectory segments. Figure 5 shows concentrations of null segments as graduated symbols in a map, focusing on the Brazilian south and southeast

regions. Naturally, the sites of major Brazilian universities concentrate null segments, in cities such as São Paulo, Rio de Janeiro, Porto Alegre, Belo Horizonte and Campinas. This result does not imply that the researchers tend to work close to their origins; an analysis on that is presented in Section 4.7.

In the next subsection, we show the classification of segments according to the educational stages in the researcher's trajectory.

4.4 Education degree

Looking now at the spatial distribution of segments according to education degree, we observe that the highest incidence of stages outside Brazil occurs in post-doctorate work (Table 3). Naturally, as the level of specialization increases, the more people go abroad for their studies. This trend is shown in Figure 6. For the master's degree, the destinations concentrate in the American East coast and in Western Europe. Ph.D.-related destinations are more varied, and more institutions serve as destination. In the INCT database, doctoral studies abroad are 3.6 times more frequent than master's degrees obtained abroad. Post-doctoral trajectories are the most internationalized, and the map shows that post-doctoral stages are held in an even greater variety of institutions. The number of post-docs held abroad is about twice the number of Ph.D.s obtained abroad. Post-docs in Brazil comprise 38% of the total, whereas almost 80% of the Ph.D. work is conducted in Brazilian institutions. This is to be expected, since post-doctoral stages are meant as an opportunity for the researcher to expand on his cooperation network and to interact with top research groups in his area, wherever they are located. The high concentration of Ph.D. degrees obtained in Brazil shows that Brazilian institutions are capable of educating most of the country's researchers, at least from the INCT point of view.

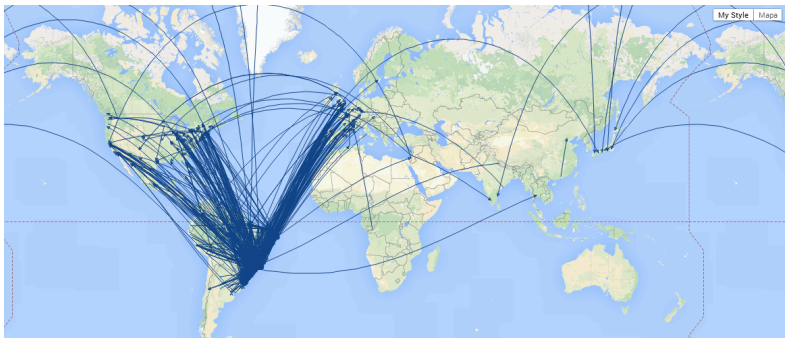
Figure 5. Null trajectory segments

Within Brazil, similar patterns can be seen. Figure 7 shows displacements for master's, Ph.D. and post-doc stages, indicating a concentration of destinations in southeastern institutions. These maps concentrate all links to or from a state in the state's centroid, so that the most important flows can be perceived. Notice the important

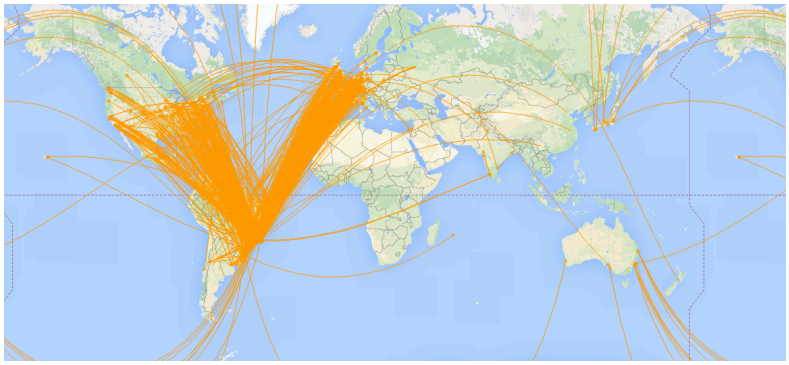
link in the master's degree map, towards Pará. These trajectories indicate a concentrated interest for the environmental researchers.

Table 3. Distribution of segment destinations per degree

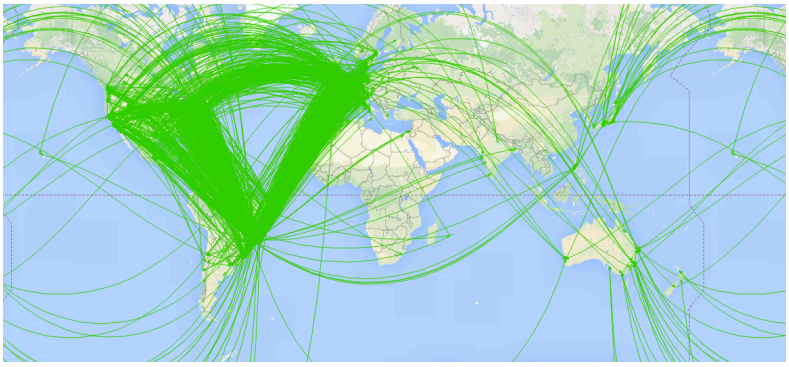
	Brazil	USA	Europe	Other
Bachelors	514	4	22	14
Masters	4,784	117	139	68
Ph.D.	4,570	351	697	119
Post-doc	1,486	1,056	1,081	279



(a)

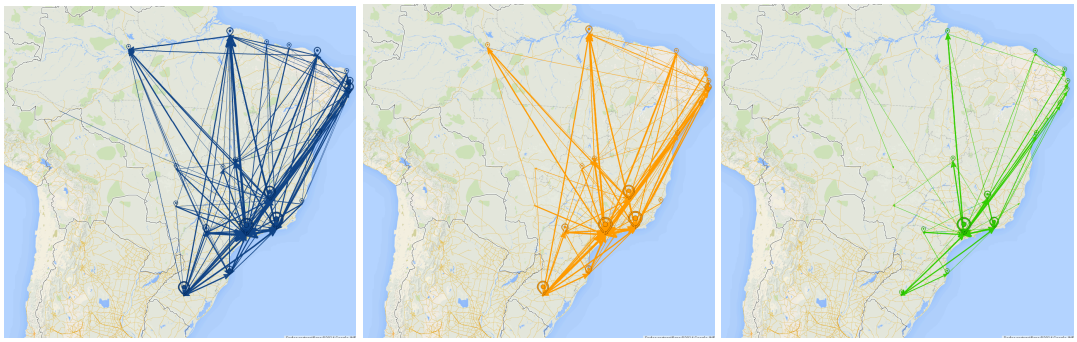


(b)



(c)

Figure 6. Trajectory segments for (a) master's, (b) Ph.D., and (c) post-doc degrees

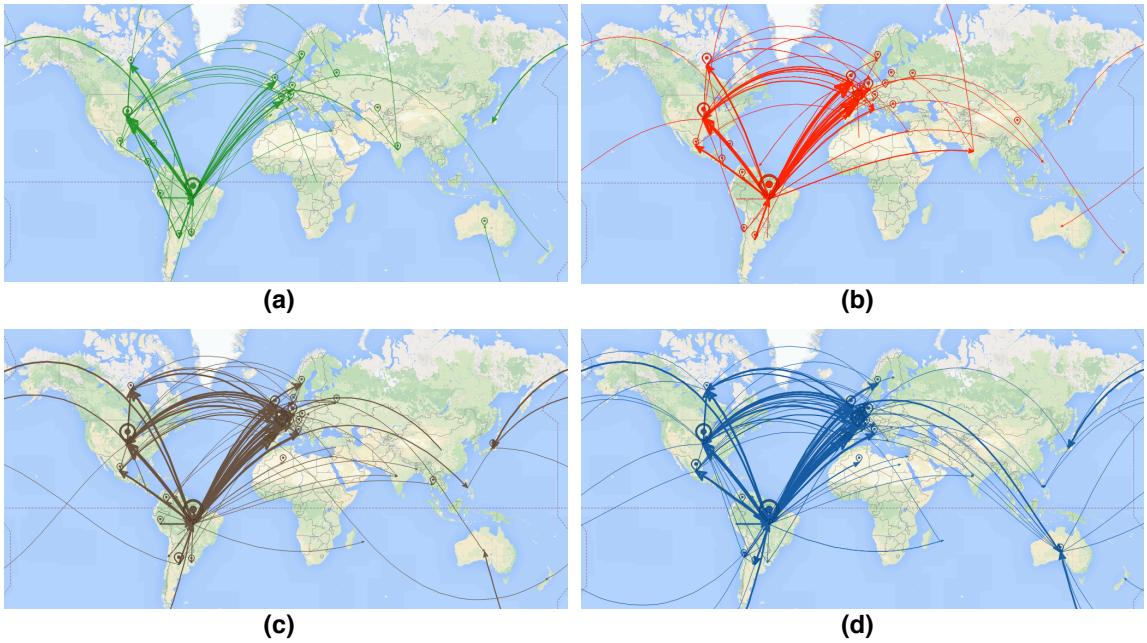


(a) (b) (c)

Figure 7. Trajectory segments for (a) master's, (b) Ph.D., and (c) post-doc in Brazil

4.5 Variations along time

Figure 8 shows the mobility filtered by decades. In these maps, all incoming or outgoing links for a country are concentrated in a single node within that country, and indicating the number of coincident trajectories using line thickness. Icons and icon sizes indicate the number of null trajectory segments. We observe that, although the general pattern apparently does not change along time (i.e., the attractive destinations continue to be attractive), the intensity (reflected in the thickness of the lines) varies significantly, reflecting the growth in the number of researchers along time (Table 4). The decades from 1980 to 2000 concentrate most of the INCT researchers. Researchers that studied in the 1970s are scarcer, since many of them would be near the age for retirement by now.



(a) (b)

(c) (d)

Figure 8. Trajectory segments along time: (a) 1970s, (b) 1980s, (c) 1990s, (d) 2000s

We can see an evolution in the number of links leading to Europe. The trajectories involving countries in Asia and Oceania also increases, but that can be

associated with the increase in the number of researchers. The trajectories between USA and Europe appear throughout all the period, and show that this behavior is not recent, even though most links are related to post-doctoral work.

Table 4. Number of trajectory segments per decade (end year)

Decade	# Segments
1950	12
1960	119
1970	871
1980	2338
1990	5205
2000	6004
2010	752

4.6 Variations according to INCT research area

The visualization tool can also filter trajectories based on the research area associated with each INCT. This allows us to verify the preferred destinations for academic degrees according to the field of knowledge. Figure 9 shows three peculiar areas in this respect. With maps that exclude trajectories traversed by a single researcher. Humanities researchers (Figure 9a) seem to prefer European destinations, France in particular. Engineering and IT (Figure 9b) seek the USA and Europe, with a major drive towards London. Health/medical researchers (Figure 9c) have a wider variety of destinations, with an emphasis on the USA's east coast, various locations throughout Europe and a number of Asian destinations.

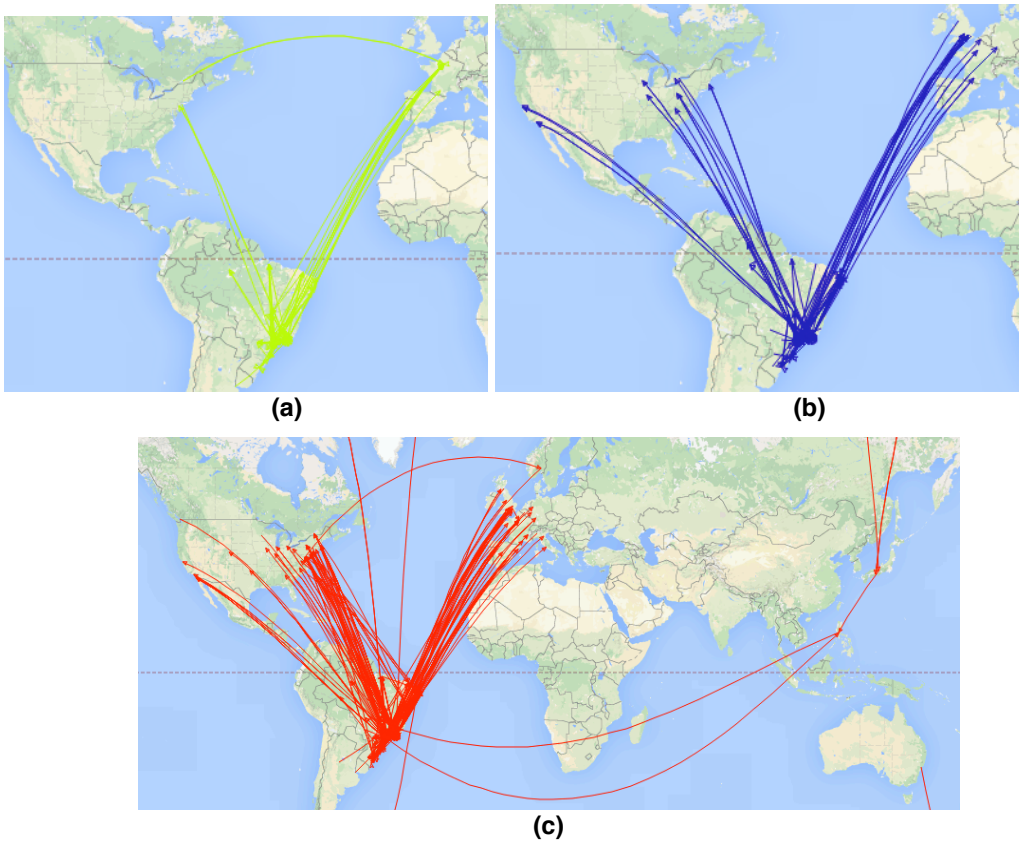


Figure 9. International destinations according to INCT research area: (a) humanities, (b) engineering/IT, and (c) health/medical

4.7 Start and end points of trajectories

Table 5 shows mobility data among Brazilian states. Column *From* indicates the number of researchers whose first trajectory node is in that state. *i.e.*, the state in which they obtained their first degree, which we assume to be close to home. Column *Stay* indicates how many of the first group ended up working in the same state. Columns *Out* and *In* respectively indicate the number of researchers that have gone to work elsewhere, and the number of researchers that move in to the state. The final column, *Work total*, indicates the number of INCT researchers currently employed by institutions in that state. Overall, 41% of the researchers end up working in a state that is different from the one in which they obtained their bachelor's degree. Table 5 also shows that some states notably attract researchers whose trajectories start elsewhere, especially Mato Grosso do Sul (91% of the researchers come from somewhere else), Sergipe (86%), and the Federal District (73%), excluding the states with a very small number of researchers. Other states are notable for exporting their graduates for work in other states, as in the case of Espírito Santo (69% of the graduates of Espírito Santo work in other states), Federal District (63%) and Piauí (56%). On the other hand, relatively few researchers from Amazonas, Pará and Rio de Janeiro leave the state (16%, 26% and 27%, respectively).

Table 5. Mobility of researchers among Brazilian states

State	From	Stay	Out	In	Total
Acre	2	1	1	8	9
Alagoas	18	12	6	27	39
Amapá	-	-	-	5	5
Amazonas	50	42	8	106	148
Bahia	142	91	51	86	177
Ceará	159	110	49	40	150
Distrito Federal	117	43	74	115	158
Espírito Santo	29	9	20	11	20
Goiás	52	23	29	34	57
Maranhão	17	8	9	10	18
Mato Grosso	30	17	13	39	56
Mato Grosso do Sul	25	11	14	107	118
Minas Gerais	654	389	265	178	567
Pará	179	133	46	174	307
Paraíba	134	65	69	44	109
Paraná	259	126	133	102	228
Pernambuco	231	125	106	80	205
Piauí	18	8	10	10	18
Rio de Janeiro	949	692	257	269	961
Rio Grande do Norte	94	58	36	71	129
Rio Grande do Sul	540	314	226	67	381
Rondônia	1	-	1	16	16
Roraima	2	2	-	16	18
Santa Catarina	118	68	50	123	191
São Paulo	1,588	1,004	584	587	1,591
Sergipe	15	7	8	42	49
Tocantins	1	1	-	1	2
Outside Brazil	367	9	358	55	64
Total	5,791	3,368	2,423	2,423	5,791

5 Discussion and Future Work

This work shows visualization techniques applied to academic mobility data, extracted from the records of the researchers' Lattes CVs. The data source is highly reliable, being supplied and curated by the researchers themselves, but the process of obtaining and preparing such data for visual analysis is not yet straightforward. We created a Web application in which users can interactively manipulate visualization parameters, data

filters and other controls to analyze the mobility of researchers according to many different perspectives, from the individual trajectory to regional grouping.

In our visually-enhanced exploratory analysis of the data, we observed the prevalence of Brazilian destinations in the various steps of the researchers' education, and showed preferences as to international stages. Regarding the degree of education, we observed a tendency towards seeking foreign positions in the latter stages, especially for post-doc. Brazilian institutions handle most of the degrees, with some mobility between states and a concentration in the Brazilian southeast region, especially in São Paulo state. We found distinct mobility pattern variations between the various research areas, and further study could help determining more about the preferred institutions. We also observed concentrations of researchers of a given research area in some regions of the country – the most notable example is the environmental field, in which many researchers move to institutions closer to the Amazon. In the time-based analysis, we noticed that the prevalence of education degrees within Brazil is growing, but the participation of international stages is important.

Future work includes expanding the database to a larger and more variety set of researchers, and improving the visualization tool as to increase the variety of visual parameters that can be controlled by the user. We also emphasize the possibility of using such tools for other kinds of data, such as students in the Brazilian government's Science without Borders program, or migration demographics from the decennial census or from the Ministry of Education's student census.

Acknowledgments

Authors acknowledge the support of CNPq (308678/2012-5), FAPEMIG (CEX-PPM-00518/13), and CAPES, Brazilian agencies in charge of fostering research initiatives.

References

Câmara, G., A. M. V. Monteiro, S. Druck and M. S. Carvalho (2004). Análise espacial e geoprocessamento. Análise espacial de dados geográficos. S. Druck, M. S. Carvalho, G. Câmara and A. M. V. Monteiro. Brasília (DF), EMBRAPA.

Etienne, L., T. Devogele and A. Boujou (2012). Spatio-temporal analysis of mobile objects following the same itinerary. Advances in Geo-Spatial Information Science, CRC Press. **10:** 47-57.

Haining, R. (2003). Spatial data analysis: theroy and practice, Cambridge University Press.

IDEA Consult (2010). Study on mobility patterns and career paths of EU researchers. Brussels, Belgium, European Commission, Research Directorate-General.

Laender, A. H. F., M. M. Moro, A. S. Silva, C. A. Davis Jr, M. A. Gonçalves, R. Galante, A. J. C. Silva, C. A. S. Bigonha, D. H. Dalip, E. M. Barbosa, E. N. Borges, E. Cortez, P. Procópio Jr., R. O. Alencar, T. N. C. Cardoso and T. Salles (2011). CiênciaBrasil - the Brazilian portal of science and technology. Seminário Integrado de Software e Hardware (SEMISH). Natal (RN), Brazil, Sociedade Brasileira de Computação (SBC)**:** 1366-1379.

Schich, M., Song, C., Ahn, Y.-Y., Mirsky, A., Martino, M., Barabási, A.-L. Helbing, D. (2014). "A network framework of cultural history." Science **345**(6196): 558-562.

Spaccapietra, S., C. Parent, M. L. Damiani, J. A. Macedo, F. Porto and C. Vangenot (2008). "A conceptual view on trajectories." Data & Knowledge Engineering **65**(2008): 126-146.

Van Bouwel, L. A. C. (2010). International mobility patterns of researchers and their determinants. Sumer Conference 2010 on Opening Up Innovation: Strategy, Organization and Technology. Imperial College London Business School**:** 1-26.

Visualizing the Quality of GNSS Multivariate Data

Bruno César Vani[1], Ivana Ivánová[2], João Francisco Galera Monico[2], Milton Hirokazu Shimabukuro[3]

[1]Programa de Pós-Graduação em Ciências Cartográficas – Faculdade de Ciências e Tecnologia da Universidade Estadual Paulista (FCT/UNESP) – Rua Roberto Simonsen, 305 – 19060-900 – P. Prudente – SP – Brazil

[2]Departamento de Cartografia – FCT/UNESP

[3]Departamento de Matemática e Computação – FCT/UNESP

brunovani22@gmail.com, {i.ivanova, galera, miltonhs}@fct.unesp.br

Abstract. *This paper presents aspects related to the definition, management, and communication of data quality with application to a multivariate data set from Global Navigation Satellite System (GNSS). We stated about the quality in a context of scientific research with GNSS data presenting quality elements that can be applied to scientific research. A case study focusing on time varying quality elements is presented, where interactive data quality visualization schemes are applied to visualize positional accuracy and attribute accuracy varying on time, allowing the users to perform queries. We highlight the main features of the implemented visualizations and conclude with suggestions for future applications.*

1. Introduction

Global Navigation Satellite System (GNSS) allows one to obtain positioning through reception of signals from the satellites of the corresponding system, like GPS, GLONASS and Galileo. The positioning can be achieved with just one system or in combination of more. Currently, GNSS applications extend far beyond positioning and navigation, since observations from GNSS may support research on Aeronomy, Meteorology, and other areas.

Even restricting for the main target of GNSS – the positioning – various methods to provide the position can be applied. Therefore, starting with data acquisition, continuing with data processing and obtaining the resulting position, distinct configurations can be established. These configurations can include, for example, the models to correct the effects that distort the GNSS signals and methods for adjusting the observations. Similarly, other research areas that apply GNSS data can have different configuration methods to reach the desired results. In this context, thinking about the quality of the obtained results may include several ramifications due to the various possibilities available to apply GNSS data.

Harding (2013) presents a central question in the context of data quality: how fit are the data for specific use and user? To answer this question, users need information about the datasets quality. Since it is provided, they can judge if the data is relevant for a specific use context.

In this paper we present about providing and communicating data quality information about GNSS multivariate data set in a context of scientific applications. The discussion covers data quality applied to researchers of GNSS scientific applications, in contrast of the general public who use GNSS positioning technology to navigate. The objective of this paper is to emphasize the importance of the data quality information with a practical case study. We selected a delimited context that covers a GNSS application of standard GPS positioning and also a related application concerning ionosphere monitoring, both being supported by a network of GNSS receivers. We also introduce aspects about the management of time varying data quality information, as well as data quality visualization and interactive manipulation schemes as efficient ways for communicating quality information to the users.

2. GNSS Applications

Nowadays, GNSS technology is ubiquitous. Receivers providing position in real-time and supporting navigation are embedded on cars, watches, tablets and other mobile devices. GNSS technology is also applied in several other professional activities. For example, one can mention the precision agriculture, where high accuracy GNSS positioning is applied allowing the farmers to reach the better application of resources in field plants, and consequently fetching financial advantages. Many other examples could be mentioned, such as Georeferencing, Meteorology, Reflectometry and others, having all of them benefits from the GNSS technology.

In addition of applications that use the GNSS positioning technology, GNSS is also important for scientists. Research related to GNSS technology may follow several ramifications. One example is the research related to improve the positioning accuracy provided by GNSS technology, as proposed for instance in Luo et al. (2008) and Aquino et al. (2011). Furthermore, as GNSS signals propagate by the Earth's atmosphere, the propagation properties of the signal allow investigations on other research areas, i.e., GNSS signals are used not only to obtain positioning. For example, GNSS measurements can support Meteorology on weather forecasting (Sapucci et al. 2009), Aeronomy applications, such as ionospheric studies (Muella et al. 2011), and Reflectometry for near surface applications (Nievinski & Larson, 2014). All of these research use GNSS data. Therefore, providing data quality information is an essential aspect since it may allow different researchers to choose data of desired quality then ensuring the reliability of the derived results according to the requirements of the different research areas.

2.1. Scientific GNSS Applications

From a technical point of view, GNSS are constituted of satellite constellations that allow one to get position based on range measurements between the antennas of receiver and satellites. Those range measurements are based on the propagation time of the electromagnetic signal from the orbital satellite to the ground receiver, and are well known on the literature as the GNSS observables (Seeber, 2003).

Apart from GNSS observables that provide positioning, some special receivers or add-ons can also provide different parameters like climatological indices and signal

processing metrics. Similarly, one can use the combination of these GNSS based parameters in conjunction with external information in order to produce other GNSS derived measurements. For this reason, in this paper, we denote GNSS multivariate data to refer to any kind of data that can be obtained, calculated or derived from GNSS measurements, then including the GNSS observables that provide positioning and all derived metrics or computed values.

General workflow presented in Figure 1 denotes data usage in scientific GNSS applications. The workflow states the data acquisition and the achievement of the results being linked by the data processing step. It is necessary to clarify that distinct workflows could be established covering different applications or points of view.

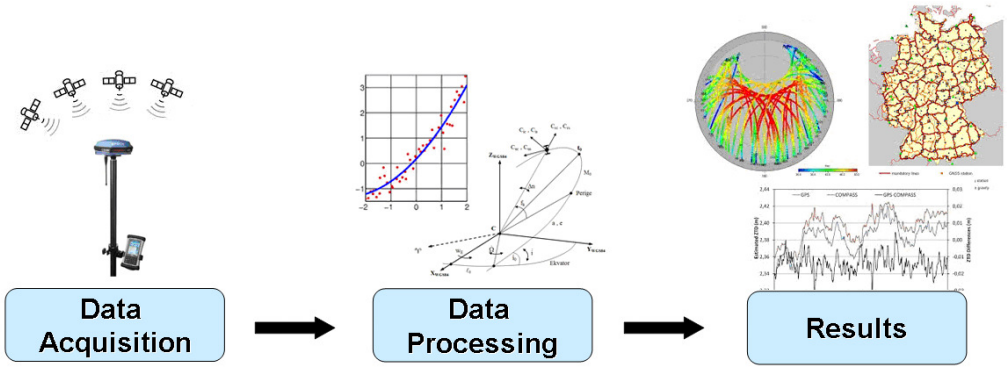


Figure 1. Scientific GNSS application workflow.

On the data acquisition, in spite of user's applications - where the own user's receiver collects the data - many research on GNSS are based on data collected by third part agencies/networks. One example is the network managed by the International GNSS Service (IGS), which is composed by receivers spread across the whole Earth surface. The receivers track continuously signals from the GNSS systems; these data are stored, constituting therefore a useful data repository that can be used on several research related to GNSS. In order to interchange data between different receivers, a common data format has been established since 1989: the Receiver Independent Exchange Format - RINEX (Gurtner et al. 2007). The data tracked by any station belonging to the IGS network can be downloaded in RINEX format by interested users. Other standard file formats are applied to GNSS product – like IONEX, SINEX and SP3 –, as well as standards for transmission of GNSS data in different transmission medium. These file formats specifications can be found at the IGS website (http://igscb.jpl.nasa.gov/igscb/data/format/).

Data processing is carried out after the data acquisition (see Figure 1). The data processing configuration may differ depending on the methods of GNSS positioning used. For example, one can mention the Standard Positioning Service (SPS) from GPS observables. SPS assumes the basic usage of GPS, which means the usage of a single receiver to get absolute position with accuracy from about 5 to 15m. Another example is given by the Precise Point Positioning (PPP), which allows obtaining centimeter accuracy by applying corrections for all systematic errors that affect the GNSS observables together with precise orbits and clocks. There are also other methods that

are based on differential corrections and relative positioning provided by base receivers in order to be as much accurate as possible (Seeber 2003). For the first case we have the differential positioning with metric accuracy while in the relative positioning using carrier phase, the accuracy is in the cm level.

Finally, the result of data processing is the position or the other desired output, like an ionospheric index (see Figure 1). These results are achieved from different mathematical models those can be applied during data processing, allowing one to obtain certain result.

3. Data Quality Review

Data quality is defined as "totality of characteristics of a data that bear on its ability to satisfy stated and implied needs" (ISO 19101:2002). The data quality information allows users to assess the fitness of the data for a specific purpose. In addition, it is possible to estimate the quality of derived products from a dataset (Goodchild & Clarke 2002). The aspect of allowing users to assess the fitness of the data for a specific purpose can be applicable in research using GNSS data by properly using data quality evaluation metrics.

GNSS data contains uncertainty, which arises from measurement errors, observation operator errors, misleading computation, data corruptions, etc. Consequently, processing uncertain data will propagate the uncertainty to subsequent levels (Williams et al. 2009). Therefore, the concept of data quality can also refer to the degree of uncertainty of the data (Zaixian et al. 2006).

We represent uncertainty in positions achieved with GNSS technology with positional accuracy, which is defined by ISO as "the degree of accuracy of positions within a spatial reference system" (ISO, 2013). If the ground truth of an expected position is available, the positional accuracy can be expressed with classical statistical measures, such as the root mean square error (RMSE). Another indicator is the absolute positional error for a given position, that can be estimated by comparing the surveyed coordinates (for instance X, Y and Z) against the known coordinates (X', Y' and Z'), where one can obtain the resulting tri-dimensional error as follows:

$$Error_{3D} = \sqrt{(X'-X)^2 + (Y'-Y)^2 + (Z'-Z)^2} \qquad (1)$$

Considering the GNSS derived metrics - such as ionosphere monitoring indices based on GNSS observations - for the cases where the expected value for the attribute is available, the RMSE can also be applied, now referring to the quality element of attribute accuracy, i.e., how correct these attribute values are (Hacklay, 2010).

When the expected value for the attribute is not available, the classical statistical measurement of the Standard Deviation (SD) can be used to get an indicator of the precision of the variables. Considering a GNSS derived metric x been computed n times, with an average value of $\bar{x}$, the SD can be computed as follows:

$$SD = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n-1}} \qquad (2)$$

Other data quality element applicable to our context is the completeness. With completeness, we measure the absence of data, i.e., how many data is missing (ISO 19157:2013). Considering that GNSS data is tracked by networks over time, there is a challenge to manage data quality elements (positional accuracy, attribute accuracy and completeness), because they are also time varying.

Other elements of quality described and discussed in (ISO 19115:2003, ISO 19157:2013, Hacklay 2010, Harding 2013) are also applicable for multivariate GNSS. These elements may cover possible requirements of researchers of scientific GNSS applications using data provided by GNSS networks. We denote these elements as descriptive (containing plain text information) data quality elements :

- Provenance: describes how the GNSS data is created containing description of the GNSS receiver use for data acquisition and the computations performed in the case of GNSS derived parameters;
- Temporal validity: the time or the time range to which the measurement corresponds;
- Resolution: temporal and spatial resolution of the measurement;
- Availability and access: describe how one can find and access the data;
- Usage: describe the basic usage of the data.

3.1. Data Quality Visualization

Data visualization provides effective and attractive means to process information in massive datasets, as well as it provides communicate data quality to the users (Goodchild & Clarke 2002. Zaixian et al. (2006) highlights the importance of designing visualizations that cover not only the data, but also its quality. In that case, appropriate data quality visualization schemes can allow one to get more insight to the data and its quality in a straightforward way. Providing data quality information visualization schemes can be reached by managing data quality as a new attribute on the visualization process or trying to superimpose quality information on the current visualizations.

One example of managing quality elements as new attributes on visualizations is available at the SIRGAS (Sistema de Referencia Geocentrico para as Americas) website (available at http://www.sirgas.org/index.php?id=206). A map identifies the stations position on the territory, as illustrated on Figure 2 a). By selecting one station, the positional accuracy information over time is presented on a triad of scatter plots - Figure 2 b). Users can interactively select the stations to check its coordinates' behavior along the time.

4. Architecture and Technological Components of the System for GNSS Data Quality Visualization

The objective in our case study is to manage the quality of the results derived from GNSS data processing. In this section, we present the architecture of a system for visualizing the quality of GNSS data.

The main components of the system are presented on Figure 3. The GNSS constellation is a set of several GNSS systems - GPS, GLONASS and Galileo - with satellites broadcasting signals that propagate through the Earth atmosphere. These signals are tracked by GNSS receivers on the ground segment and are organized in

networks administered by several institutions. Receivers transform signals into data that can be stored in their own format; those receivers for ionosphere monitoring also calculate related parameters. Data are transmitted using FTP protocol to the central data repository where they are written in standard format files before being stored in a database. Using Data Base Management System (DBMS) to organize data makes easier the access by end user. This information is managed by the implemented system and displayed through a web interface which implements dynamic and interactive visualization schemes. We apply quality evaluation metrics presented in section 3 to provide the data quality information to the users. This allows users searching for data of desired quality and selecting it for further use in their own applications. More details are presented with the case study in Section 5.

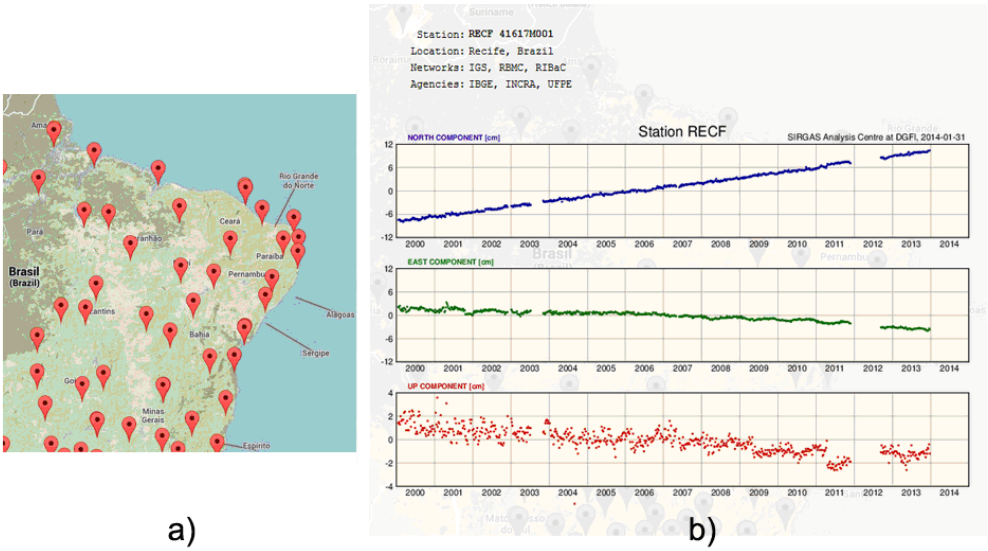


Figure 2. Example of displaying positional accuracy information over time given at the SIRGAS website, available at http://www.sirgas.org/index.php?id=206. The map identifies the stations position on the territory - a). By selecting one station, positional accuracy information over time is presented on a triad of scatter plots - b).

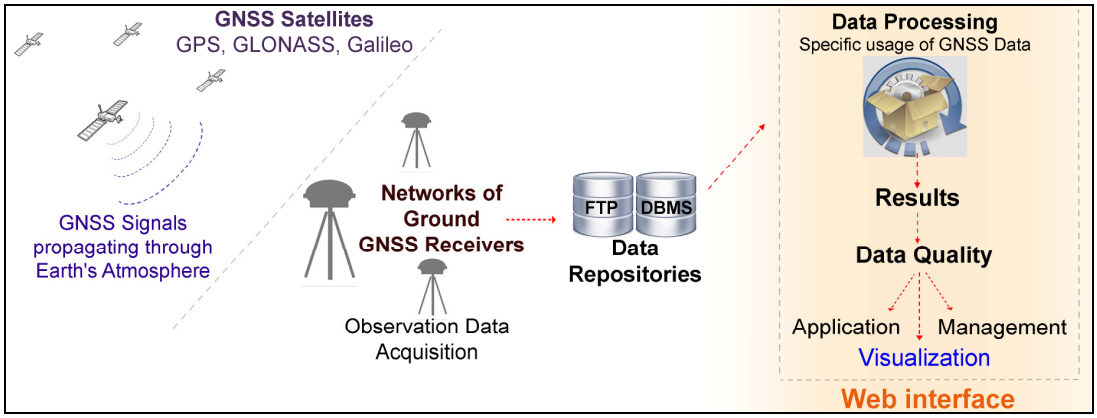


Figure 3. Organization of software and hardware components and data flowing through them.

5. Case Study

The objective of our case study is presenting the definition, management and communication of data quality information on scientific GNSS applications. We focus on the communication of positional accuracy and attribute accuracy in a time varying context. The results presented in this section were implemented within the software ISMR Query Tool – a web application for visualization and mining of GNSS derived data with emphasis on ionosphere monitoring data. The acronym ISMR denotes "Ionospheric Scintillation Monitor Receiver", a special type of GNSS receiver that provides ionosphere monitoring parameters. The system is accessible at http://is-cigala-calibra.fct.unesp.br/.

5.1. Representing uncertainty in positions achieved from GNSS

The subject of our case study is the GNSS data available from the Brazilian Network of Continuous Monitoring of GNSS Systems (RBMC – acronym in Portuguese). This network is managed by the Brazilian national geographic and statistics institute (Instituto Brasileiro de Geografia e Estatística - IBGE). The stations' configuration of RBMC – that provide RINEX data in real time or post-processed – is presented on Figure 4.

Figure 4. Stations configuration of RBMC. Blue points: real time stations; black points: post-processed stations. Image from IBGE website, available at <http://www.ibge.gov.br/home/geociencias/geodesia/rbmc/rbmc_est.php>.

In our case study, RINEX data for each RBMC station at a given day were processed through the PPS on-line software, which performs Standard Point Positioning with GPS data. The results achieved by the software are the position (X, Y and Z) and associated precision (σ_x, σ_y and σ_z), acquired in each minute of the day. To obtain a

manageable way to manipulate the results, the positioning solutions provided by the software were inserted in a database managed with PostgreSQL DBMS.

The positional error at a given epoch was estimated by comparing the processed output coordinates against the known coordinates (X', Y' and Z') of the stations of the network, obtaining the resulting 3-dimensional error as presented by Eq. 1. One can also obtain the 3-dimensional standard deviation – another quality indicator of the positioning achievement – as follows:

$$SD_{3D} = \sqrt{(\sigma_X)^2 + (\sigma_Y)^2 + (\sigma_Z)^2} \qquad (3)$$

The interactive visualization scheme at a selected epoch is presented on a map as shown in the example of Figure 5. The user can choose between the positional accuracy indicators available ($Error_{3D}$ or SD_{3D}). Red circles are drawn over the station coordinates where the radius' size of each circle is mapped according to the value of the chosen indicator. In order to assure a user-controlled visualization scheme, the user can specify manually the scale range of the chosen parameter as well as the expected range of the representative circles. The circles are drawn in a linear scale according to the specified range values.

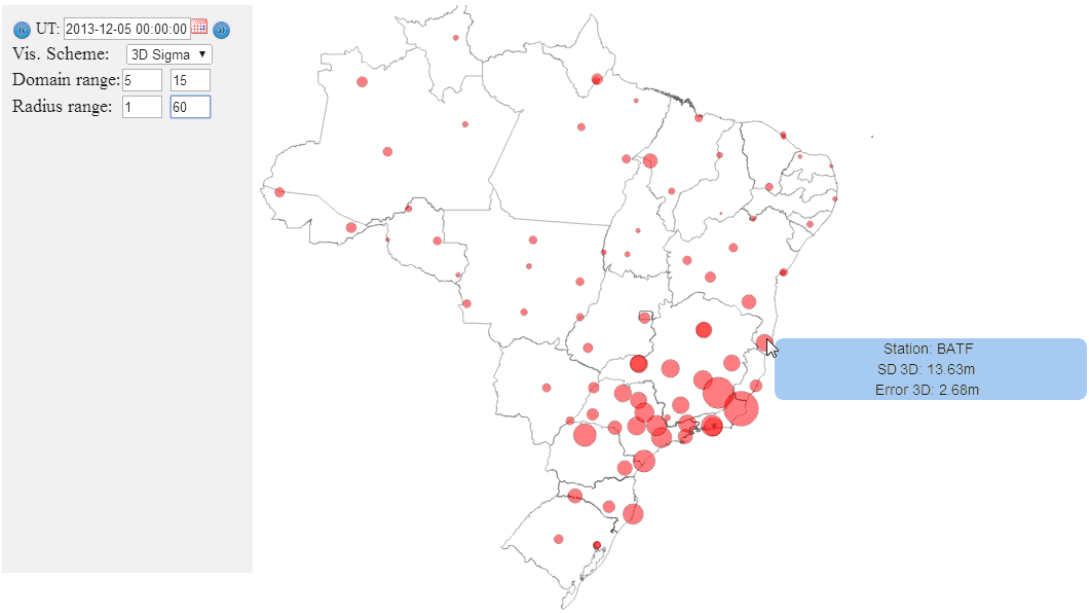


Figure 5. Interactive visualization scheme for Sigma 3D for the positioning solutions for 2013-13-05, 0h (UT).

The interactive visualization can be zoomed in our out, the scales can be changed, and interactions with the mouse give more details about a selected station.

Next to the representation of the results achieved at a single epoch, another possibility is to represent the aggregated positional accuracy information, such as average or standard deviation over a specific time interval. One example is shown in Figure 6, where the average of values of 3D error for the whole selected day is represented. The domain range of the values of 3D error was adjusted from 5 to 10 meters, i.e., the user is expecting that the 3D error has amplitude from 5 to 10 meters.

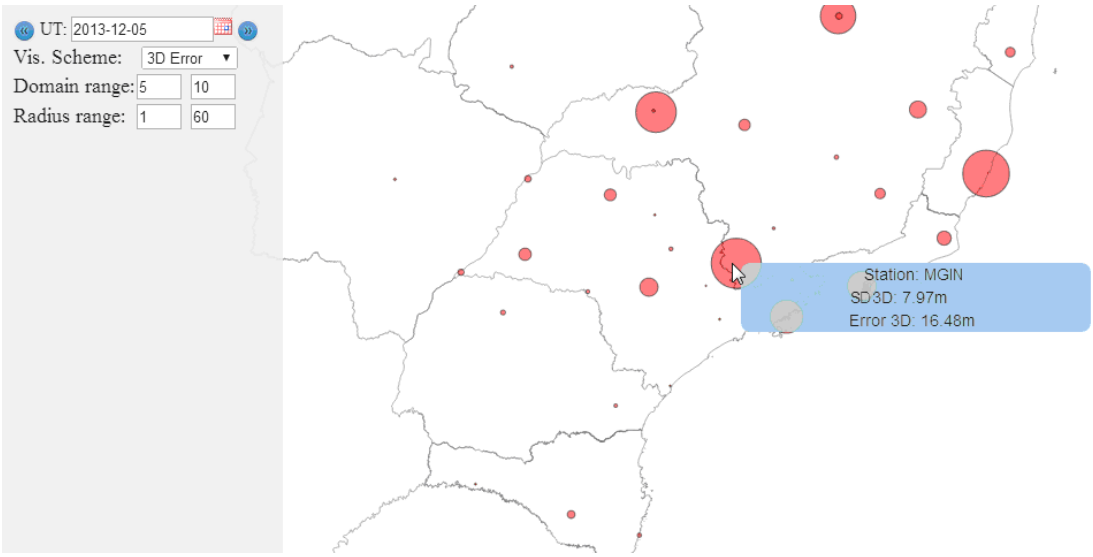


Figure 6. Interactive visualization scheme for averaged Error 3D for all the positioning solutions of 2013-13-05.

5.2. Representing uncertainty in attribute derived from GNSS data

We choose the CIGALA/CALIBRA network's dataset to provide an example for definition, management and display of uncertainty in attributes. This network – that is managed by FCT/UNESP in conjunction with many partners – is currently composed by eleven receivers spread across the Brazilian territory (see Figure 7).

Figure 7. Distribution of stations of CIGALA/CALIBRA Network.

The receivers in CIGALA/CALIBRA network provide derived metrics based on GNSS observations. An example of such observation is the ISMR data, which consists of a subset of several parameters that can be extracted every minute. Moreover, besides the GNSS observables available via the RINEX format, the CIGALA/CALIBRA

Network provide ISMR files, and both can be used by researchers of GNSS scientific applications.

ISMR file contains indices that describe ionospheric activity, as well as additional signal properties metrics those could characterize other effects, like noisy or reflections on the GNSS signals. The ISMR data provided by CIGALA/CALIBRA network support several researches in monitoring and modeling the South American ionosphere.

When dealing with such kind of ionospheric monitoring data, one can use the Ionospheric Pierce Point (IPP) representation in order to visualize the projection of the monitoring parameter over a map. The IPP is a virtual point where the signal passes through a layer of about 350 Km above the Earth surface. The IPP scheme is presented in Figure 8.

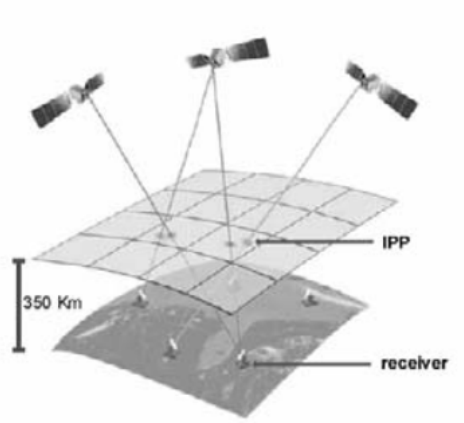


Figure 8. IPP representation. Image from Rezende et al. (2007).

Differently of the example presented on subsection 4.1, in this example we do not store the quality information in a database. Instead, the uncertainty is represented as an attribute (like an average or standard deviation) that can be computed dynamically and rendered with a map server engine. Therefore the result is shown over the Brazilian map with customizable grid resolution (with regular patterns in time and space) and output parameters.

One example is presented in Figure 9. The standard deviation of one selected attribute of ionosphere monitoring (the S4 scintillation index achieved from GNSS observations achieved by receivers of CIGALA/CALIBRA network) is represented for a period of one month (January 2014) with the IPP map, where users can identify an area of more frequently varying quality. On the example presented in Figure 9, values of higher standard deviation are mainly on the bottom of the station. Furthermore, one can also choose different time intervals to get the desired feature on the time varying domain.

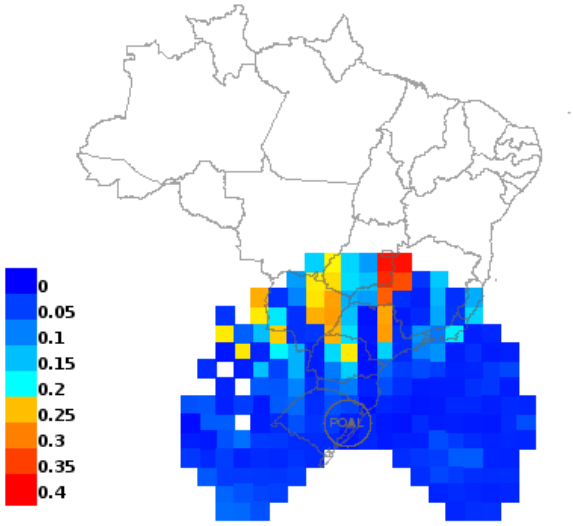


Figure 9. Example of attribute uncertainty visualization with IPP map: standard deviation of S4 ionosphere monitoring index on January, 2014 (data collected at the station POAL - Porto Alegre/RS, Brazil).

6. Conclusions and Future Work

In this paper, we discussed a system for visualizing quality information in the context of scientific applications with GNSS multivariate data. We assumed a general application workflow of data acquisition, data processing and results achievement, and defined quality elements applicable in this scenario.

We presented a system for management and visualization of positional accuracy and attribute accuracy of GNSS data for scientific applications. and demonstrated this system with a small case study. With our system researchers from different areas can visualize quality GNSS data with interactive data visualization schemes.

Our future work includes definitions and implementation of other relevant data quality parameters and improvements related to the usability of the quality visualization schemes. More possibilities to perform queries on data quality information will be explored, for instance by introducing uncertainty information into the dataset allows performing queries and find data of desired quality. In addition, the UncertML (Williams et al. 2009) concept could be applied in order to ensure a proper communication of uncertainty that could allow interoperability between systems.

References

Aquino, M., Monico, J. F. G., Dodson, A. H., Marques, H., De Franceschi, G., Alfonsi, L., Romano, V. & Andreotti, M. (2009), 'Improving the GNSS positioning stochastic model in the presence of ionospheric scintillation'. Journal of Geodesy, 83(10), 953-966.

Goodchild, M. F. & Clarke, K. C. (2002). 'Data quality in massive data sets'. In Handbook of massive data sets (pp. 643-659), Springer US.

Gurtner, W. & Estery, L. (2007), 'RINEX-The Receiver Independent Exchange Format-Version 3.00', Astronomical Institute, University of Bern and UNAVCO, Bolulder, Colorado.

Harding, J. L. (2013), 'Data Quality in the Integration and Analysis of Data from Multiple Sources: Some Research Challenges', ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 1(1), 59-63.

ISO, 2001, 'ISO19101:2001 Geographic Information – Reference model', International Standards Organisation.

ISO, 2003, 'ISO19115:2003 Geographic information – Metadata', International Standards Organisation.

ISO, 2013, 'ISO19157:2013 Geographic information – Data Quality', International Standards Organisation.

Luo, X., Mayer, M., & Heck, B. (2008), 'Improving the stochastic model of GNSS observations by means of SNR-based weighting', Observing our Changing Earth (pp. 725-734), Springer Berlin Heidelberg.

Muella, M. T. A. H., de Paula, E. R., Mitchell, C. N., Kintner, P. M., Paes, R. R. & Batista, I. S. (2011), 'Tomographic imaging of the equatorial and low-latitude ionosphere over central-eastern Brazil', Earth Planets and Space 63(2), 129.

Nievinski, F. G., & Larson, K. M. (2014). Forward modeling of GPS multipath for near-surface reflectometry and positioning applications. 'GPS solutions', *18*(2), 309-322.

Rezende, L. F. C., Paula, E. R., Kantor, I. J. & Kintner, P. M. (2007), 'Mapping and survey of plasma bubbles over Brazilian territory', Journal of Navigation 60(01), 69-81.

Sapucci, L. F., Machado, L. A. T. & Monico, J. F. G. (2009), 'Previsões do atraso zenital troposférico para a América do Sul: variabilidade sazonal e avaliação da qualidade', Revista Brasileira de Cartografia, 58(3).

Williams, M., Cornford, D., Bastin, L. & Pebesma, E. (2009), 'OGC Discussion Paper 08-122r2: Uncertainty Markup Language (UnCertML)'.

Zaixian, X., Shiping, H., Ward, M. O. & Rundensteiner, E. A. (2006), 'Exploratory visualization of multivariate data with variable quality', in Visual Analytics Science And Technology, 2006 IEEE Symposium On (pp. 183-190).

The Semantic Pixel

Fred Fonseca[1] Clodoveu Davis[2] Gilberto Câmara[3]

[1]College of Information Sciences and Technology – The Pennsylvania State University – 307E IST Building – University Park, PA – USA

[2] Departamento de Ciência da Computação – Universidade Federal de Minas Gerais Av. Antônio Carlos 6627 - Belo Horizonte - Brazil

[3] INPE - Brazilian National Institute for Space Research Av dos Astronautas, 1.758 – São José dos Campos – SP - Brazil

fredfonseca@ist.psu.edu, clodoveu@dcc.ufmg.br, gilberto.camara@inpe.br

Abstract. *Usually, images can be seen as sets of pixels or as fields over a reference space. While the former view allows image processing to function using pixel manipulation algorithms, the second one is closer to a wider understanding of what people perceive in an image. The pixel aspect is much closer to the measurement, to observations, while fields are closer to the semantic aspect, to the interpretation of the observations. This paper discusses some semantic challenges related to integration of image data from various sources, considering both views. Such integration is necessary, considering that soon a new generation of remote sensing satellites based on free and open data policies is expected to become operational, so researchers will have access to more data than they can handle with current techniques. We propose the integration of images from multiple sensors starting from a common point, which we call the Semantic Pixel. It will enable scientists to have access to large sets of satellite images and their metadata, regardless of source or format. The Semantic Pixel will also enable access to ancillary data, which is essential for advanced temporal analysis of forest cover dynamics, including major sets of natural resource data, such as vegetation, soil and geology maps. Other data encoded as fields, such as digital elevation models, relevant climatic variable maps, political maps and associated census data, can also fit this model.*

Resumo. *Imagens podem ser vistas como um conjunto de pixels ou como campos em um espaço de referencia. Enquanto os pixels permitem que algoritmos de processamento de imagens possam funcionar, os campos estão mais próximos do que as pessoas entendam o que seja o significado de uma imagem. A visão de pixels está muito mais próximo das medidas, das observações, enquanto os campos estão mais próximos da semântica, da interpretação das observações. Este artigo usa estas duas visões para discutir os desafios relativos a semântica na questão das integração de imagens de provenientes de fontes diversas. Esta integração é necessária já que brevemente novos satélites com politicas de dados abertos devem estar disponíveis o que levará os cientistas a terem mais dados do que eles possam efetivamente usar. Aqui nós apresentamos uma proposta de integração de imagens de fontes diversas usando como plataforma inicial um ponto comum,*

que chamamos o Pixel Semântico. Desta maneira os cientistas teriam acesso a dados e metadados de imagens independente da fonte ou formato. O Pixel Semântico também possibilitaria o acesso a dados históricos que são importantes para análises da dinâmica da vida das florestas tropicais.

1. Introduction

There is a consensus around the idea that the data needs for deforestation monitoring are so broad that the right way to approach this problem is to facilitate access to whatever data is available. There are certainly large amounts of valuable data collected for many scientific models on deforestation which are currently inaccessible, or worse, whose existence is practically unknown for potential users. In order to provide an adequate compromise between resolution and coverage as needed for forest assessment, a global forest monitoring system (Fonseca, Davis Jr. et al. 2009) would have to integrate satellite images of different kinds. For instance, a combination of up-to-date MODIS-class (250-meter), LANDSAT-class (30-meter) and HRC-class (2.5-meter or better) images would be required for many applications. Besides that, by 2015, a new generation of remote sensing satellites (LANDSAT-8, CBERS-3, Sentinel-2) is expected to become operational, based on free and open data policies. "The age of big geospatial data has come. Space agencies worldwide plan to launch around 260 Earth observation satellites over the next 15 years."(Ferreira, Câmara et al. 2014)

The motivation of this paper is to address this increasing volume of data from multiple and diverse satellites and support its use by scientists. We take a different stance from Werner Kuhn, who tries to model "the relevant information processes independently of sensor technology" (Kuhn 2009). We acknowledge the importance of Kuhn's approach but we also want to recognize the push from technology and measurement technologies, which seem to be driving the process of data collection.

We also want to address the new possibilities brought up by new and advanced database technologies, such as NoSQL array databases (Paul; Stonebraker 2010), that are enabling scientists to have large time series of remote sensing images available as a single multidimensional matrix. Conceptually, then, any remote sensing data can be referred to by matrix elements containing a value, a timestamp and a geometry description (Ferreira, Câmara et al. 2014). This way, any remote sensing data can be addressed and manipulated based on its most fundamental component (a pixel), instead of using full images or scenes. Pixel-wise data can be organized and handled with array databases such as SciDB[1].

Although the pixel is the basic unit for remote sensing images, the semantics lie at a much larger level. Themes, coverages, and relationships are all combinations of pixels across time, and combinations of pixels in themes and concepts across space. In this paper we focus on the challenges in starting from the basic unit of recording data, the pixel level, and going to the meaning of full observations represented by, for instance, time series, trajectory and coverages (Ferreira, Câmara et al. 2014). We want to know how pixel sets can be understood and become linked to higher semantic concepts. When the pure measurement-related pixel is regarded at a higher semantic

[1] http://www.scidb.org

level, it starts carrying more meaning than just a value, a timestamp and a location. It becomes the *semantic pixel*.

This paper is organized as follows. Section 2 reviews the relation between data and theories in science in order to understand the role that pixels play as raw data. Section 3 introduces the concept of the semantic pixel. Other aspects of the semantic pixel related to pure data, aggregation, and measurement are discussed in Section 4. Finally, Section 5 concludes the paper.

2. Data and Theories

In 1978, when GIS was starting, Sinton (1977) thought that the "largest groups of potential users will be those who already use geographic information in non-digital analytic formats". Today, Google Maps and many other geographic information sources are ubiquitous. Every phone has a GPS and you can ask for directions just by talking with your handheld device. Technology has a prevalence that is concerning.

Sinton argued that geographic information systems had to cater to user needs. It was a noble goal, although system and users are still fighting. The huge availability of data, spanning all the regions of the world and now also long periods of time, at a very reasonable (sometimes neglectable) cost, is starting to show that, maybe, neither systems nor users have the upper hand, and the data won after all. So, there must be solutions for a situation such as the one Sinton observed:

> *"The digital encoding of mapped thematic data can significantly affect its applicability by these potential user groups. Experienced analysts have been using traditional organizational structures for information storage and retrieval. Mapped thematic data is universally stored in a map record format. The analytic sub-system of a geographic information system will be more easily used if it reproduces these traditional record keeping formats. Neither polygon nor "bit plane" data structures use a map record format."* (Sinton 1977)

We have now a situation in which analysts have to adapt to the format of the incoming data. Our proposal of a semantic pixel tries to address some of these issues.

2.1 The Changing Priority between Data and Theory

Scientists used to think that data were the source of theories. Only direct access to the data could save and keep Science free from metaphysics. Logical positivists claimed that propositions are reducible to elementary propositions. They tried to show that elementary propositions are exclusively concerned with picturing empirical reality. They were in part motivated by Wittgensteins' work, that tried to show that empiricism could be founded on logic.

> *"The philosophy of the Viennese Circle is an empiricism established by logical methods. Briefly stated, it is established by showing that under analysis the meaning of concepts and propositions is, in every case, ultimately empirical. Propositions which are not ostensibly empirical in reference are therefore either reducible to empirical propositions or are simply nonsense".* (Weinberg 1960 p.25)

Later, philosophical discussions in the 20th century ended up showing that theories are created before data and therefore, theories drive the data collection process.

Now, with the increasing amount and availability of satellite data, is this scenario shifting in favor of data driving theory again?

We are not arguing for the importance of observations being driven by theory over technology driving the observation process. We are only acknowledging the deluge of data from satellites and proposing a way to incorporate it in our models of science. In a way, both data and theories contribute to Science's understanding of the world.

Again, we can use Werner Kuhn (Kuhn 2009) to clarify the problems with the current situation. While Kuhn considers observation to be "an information item with semantics that are independent of observation technology", we are explicitly acknowledging that today, with the large availability of current and historical satellite data, technology is playing a leading role in the generation of observations. It is a situation in which the technological focus is "putting encoding before modeling" (Kuhn 2009).

If we are really going back to data creating theories, this is an interesting change from the way Science works, usually with theory driving observations and not the other way around. Kant was the first to remind us that

> *"When approaching nature, reason must hold in one hand its principles, in terms of which alone concordant appearances can count as laws, and in the other hand, the experiments thought out in accordance with these principles. Thus reason must indeed approach nature in order to be instructed by it; yet it must do so not in the capacity of a pupil who lets the teacher tell him whatever the teacher wants, but in the capacity of an appointed judge who compels the witness to answer the questions that he puts to them"* (Kant 1996)

Later, Popper confirmed this approach in a famous footnote in The Logic of Scientific Discovery, telling us that we always gather facts with theories in mind. He says "… observations, and even more so observation statements and statements of experimental results, are always interpretations of the facts observed; that they are interpretations in the light of theories" (Popper 2002).

Thomas Kuhn (1996) and Paul Feyerabend (1993) also agree that there is, in general, no neutral set of facts that provide a framework for comparing competing theoretical perspectives. For these crucial areas of difference, the facts they would notice would not be neutral, but would have their existence only relative to a given paradigm. Thomas Kuhn insists that the notion of a deeper, and hence neutral, classificatory scheme, is illusory. All classifications of the facts are relevant to some theoretical framework, in Thomas Kuhn's view. A so-called deeper classificatory scheme would not be neutral but relative to its own theoretical perspective.

We might think that the accumulation of decades of remote sensing data and the increased availability of images might be changing the situation described above because now we have access to the pure facts again, and lots of them. But we have to remember that instruments are also supported by theories:

> *"Galileo claimed that he could 'observe' mountains on the moon and spots on the sun and that these 'observations' refuted the time honoured theory that celestial bodies are faultless crystal balls. But his 'observations' were not 'observational' in the sense of being observed by the-unaided-senses: their reliability depended on the reliability of his telescope-and of the optical theory*

> *of the telescope-which was violently questioned by his contemporaries. It was not Galileo's pure, untheoretical-observations that confronted Aristotelian theory but rather Galileo's 'observations' in the light of his optical theory that confronted the Aristotelians' 'observations' in the light of their theory of the heavens".* (Lakatos 1970)

While hardware development and a wider satellite availability makes the pixel more complex one on side, new scientific theories and a deeper understanding of the Earth and its environment seems to create more complexity on the other side. The work in this paper is a step in the direction of understanding these two sides so that data enable better scientific theories and new scientific theories keep pushing the development of new measurement technologies.

In our case, even if remote sensing development might originate from pure Physics development, in a way unrelated to the sciences that will use the data, the understanding of the use of imagery data in Science is our goal. Therefore, data collection processes and the way such data are linked to theories are matters of interest. The discussion carried out in this Section is a step in understanding how satellite data at the pixel level can be kinked to complex concepts in scientific theories.

3. The Pixel as an Information Element

When trying to integrate images from various remote sensing sources, specialists are always confronted with problems regarding the compatibility of images as to spatial resolution, spectral resolution, and time. Putting together two images from different sources potentially requires the use of techniques and algorithms such as registration, radiometric correction, noise reduction, and others, usually applied in a pixel-by-pixel basis, and often causing distortions or information loss.

In the sense that an image is a representation of part of the world, however, concepts such as resolution should have no real value. Pixels should be viewed as an algorithm's way to break a larger problem (i.e., image processing) into many smaller ones (i.e., transforming pixel values in the vicinity of a point or inside a region). Pixels are samples to a phenomenon that extends well beyond their limits, and which cannot be perceived if we regard them individually.

As a result, there should be a level of semantic description of images in which pixels are irrelevant (Koenderink 2005). Operations on images could then be specified as functions that receive one or more representations of space and certain parameters, and generate other representations, regardless of how these representations are materialized in the first place. The most important concept in this situation is space and its characteristics; if we were able to represent every single point in space and associate it with measurements, the field representation (Couclelis 1992; Câmara, Freitas et al. 1994) would be perfect. Since we are not capable of doing so, we must deal with a coarser sampling, i.e, a measurable pixel size.

Problems begin to arise when one tries to integrate two of such representations, each of which obtained with a different set of parameters (sampling, resolution), with different sensors, at different times. In order to harmonize such differences, we propose the definition of an information element that is more generic than the pixels that compose current images, and that is strictly related to a small fragment of space. Each of these small fragments, which we call a *semantic pixel* (SP) from now on, is then

associated to the various fields that are defined over its position, so that it is possible to find out information about that position from numerous sources. There should be only one SP for each fragment of space. All recorded phenomena that occur at that position are associated to the same SP, so that it is possible to select data and metadata at that position based on a set of requirements, such as a time interval, or the response to a given electromagnetic frequency range.

Each SP becomes, then, a generic reference to the many representations and measurements that are available for that position. Given a SP, it would be possible to recover the original images that include that position, as in an index, selecting among them from their metadata. In places where the information is missing or has been corrupted (for instance, when there is cloud cover over part of a RS image), algorithms or visualization could use data from other sources, also associated to the same SP. Examining SP data, users could select the attribute (or combination of attributes) which is more adequate to fulfill a given task. Furthermore, data from each SP used for integration can be traced back to its source, so that metadata apply to SP attributes, not only to an entire image. Results from combining and processing data from various sources at the SP would also be associated with provenance metadata, so that specialists could trace the outcome of analyses and processing to their origin.

One of the motivations for the creation of the semantic pixel is traceability. Instead of separately managing several images for one point in space, the SP is a representative of those images. The SP is only a link to the original and preserved images. It gives a common ground for visualization and navigation. This is especially important for remote sensing images, which can suffer from local (i.e., cloud cover, atmospheric effects) or global (georeferencing, registration) distortions, the correction of which always raises many questions.

A system using semantic pixels will have many images (raw and processed). The user has to have the final decision on what image to use in each algorithm. The SP will enable the user to do so. Different users may use the system in different ways. A scientist may want to use the raw satellite images, do all the pre-processing, segmentation and classification, apply her own algorithms and report her findings. Another scientist may use the pre-processed images already available and perform only classification and analysis using different algorithms for verification purposes. In both cases, the SP functions as both the starting point and the link between the region of study and all the available data.

The SP must also provide semantic interoperability. An application that intends to monitor deforestation needs high temporal resolution and adequate spectral resolution, but accepts low spatial resolution. On the other hand, an application that needs to measure deforestation requires a medium spatial resolution and adequate spectral resolution, and accepts a lower temporal resolution. Therefore, a deforestation monitoring pixel is different from a deforestation measurement pixel, since the sources of information are different. The SP has to provide for both cases.

Every SP corresponds to a geographic position, and is potentially related to many *content sets*. Each content set indicates one source of data for the position that corresponds to the SP. Content sets can be viewed as tables associated to each SP, in which each row corresponds to a data source and includes at least the following columns for data and metadata: source image URL, image timestamp, source image

position, satellite (vehicle), sensor, band (frequency), spectral resolution (bits), and a Boolean indicator as to the position being cloud-covered. Value data, i.e., the measurements associated to the SP, are kept in the original image, regardless of being a raw image or a processed image (for instance, the results of a classification), but can be copied to the content set for faster retrieval. Further attributes can also be included, in order to maintain provenance data for processed images. Notice that, for every new image that is included in an archive, a new content set is created for each SP that is inside the image's spatial boundaries, but the image is also stored as received.

Looking at the resulting data infrastructure, notice that the set of SPs is highly redundant as to image metadata, in a strategy also successfully employed in tools such as data warehouses. As in data warehouses, we assume that computational resources to maintain such a level of redundancy are affordable enough to economically justify the expected semantic and performance gains. The objective is to allow the user to select a single SP, or a set of SPs, and retrieve the most adequate information available for that position or region, given a set of criteria, in a case-by-case basis. On the other hand, by grouping attributes such as the source URL together, it is possible to have direct access to the source of the information related to an SP.

4. The Many Dimensions of the Semantic Pixel

In this section we add other perspectives to the understanding of what the Semantic Pixel is. We discuss the operations that create the semantic pixel using Sinton's original ideas on the operations that pixels go through in order to be usable by scientists. Then we use Wittgenstein concept of object to discuss the semantic pixel as raw data and what it means in the scope of philosophy of science, in the middle of the discussion of theories and data. Finally we talk about the importance of understanding the pixel as measurement based on the Fonseca and Martin's (2007) conceptual framework of objects and objectives.

4.1 The Semantic Pixel is an Aggregation

Since the pixel, as most field observations are related to small sections of space and time, for its use in scientific analyses it is necessary some transformations. Sinton mentions the need for aggregation and also the problems that come with it:

> *"Original field observations tend to be voluminous in their nature. Consequently it is rare that so much data is reported and made generally available. The original collectors of the information will generalize it to reduce the volume and usually attempt to make it more understandable in an abstract form. The procedures used in the generalization or abstraction of data significantly affect its utility for analytic purposes."* (Sinton 1977)

A system such as a Global Forest Information System (Fonseca, Davis Jr. et al. 2009) using the semantic pixel would address two of the concerns Sinton had. First, regarding the user's understanding of the many transformations that raw data goes through before becoming available for scientific analyses. According to Sinton, "when presented with a set of data, individual users should attempt to understand the nature of the generalization that has taken place on the original observations or measurements" (Sinton 1977). In its original proposal, in a GFIS "there is the possibility of the different actors sharing and understanding the meaning of the scientific models

explaining deforestation processes. This way information from the different sources can be used as a communication tool, in order to motivate common citizens, scientists, and the society at large, to contribute with the monitoring effort and to influence policy making and enforcement" (Fonseca, Davis Jr. et al. 2009).

GFIS would also bring aggregation to the user of the data again. With the new database techniques, users can have access to the pure pixel again, according to Gilberto Câmara, the Holy Grail of remote sensing. This way we can control the second concern mentioned by Sinton since "associated with each of these processes of generalization is the loss of certain types of detail which existed in the original observation or measurement". (Sinton 1977 p.4)

4.2 The Semantic Pixel is a Fact

In the discussions of Philosophy of Science, Wittgenstein was trying to understand the nature of analytic propositions and how they would fit with logics and at the same time be linked to the real world. For Wittgenstein,

> *"the world is the totality of independent atomic facts. An atomic fact is a fact which is not compounded out of other facts. Since facts are ultimately independent of one another all compound facts are reducible to atomic facts. Which facts are atomic and which are not cannot be determined a priori, but must, in any case, be discovered by direct inspection."* (Weinberg 1960 p.38)

It is difficult not to think of a similarity between "atomic facts" "discovered by direct inspection" and the pixels coming from satellite sensors. Are those pixels really the most basic facts about our world? Actually, another concept Wittgenstein developed, *object*, would be close to what a pure pixel is while *fact* should be related to what a Semantic Pixel is. For Wittgenstein,

> *An object is whatever can occur as the constituent of a fact. Now, if facts are taken as fundamental, and hence indefinable, an object could be variously defined, (a) It may be defined as the set of facts in which it occurs, i.e. as the set of facts which possess at least one feature of absolute similarity to one another. For example, the facts of blue colouring the sky at time t_0 and of blue colouring this book at time t_n have one feature, blue, of absolute similarity, (b) Or an object can be defined as whatever is a distinguishable element of a fact. Thus, by exhaustively enumerating all the distinguishable elements constituting a fact, it is possible to isolate all the objects composing the fact in question.*
>
> *The fact is an independent entity, for whatever dependence may mean in the strictly logical sense, it is reserved for objects, i.e. for entities obviously requiring completion. Facts, being self-sufficient, require no completion, and so are, in the logical sense, independent of one another. Objects are independent, too, in the sense that an object is not restricted to occurrence in one fact rather than another, but they are dependent in the sense that they must occur in some fact or other.* (Weinberg 1960 p.35-36)

Since the Semantic Pixel is one level away from the pure pixels, which are closer to Wittgensteinian objects, the SP it is better represented by facts. The Semantic Pixel, as a fact, is composed of objects (pure pixels).

4.3 The Semantic Pixel is Measurement

Although the pure pixel is the most basic unit for images, and it is meaningful in its own way, there is a big conceptual leap from the individual pure pixel to sets (aggregations) of pixels representing semantic concepts such as time series, trajectories or coverages.

The pure pixel's origin, more related to the measuring hardware than to its final semantic meaning, has to be understood. It has individual meaning and complexity before it becomes part of a larger semantic concept. The use of new database techniques such as array databases only reinforces this aspect. The pure pixel, coming from the measurement hardware, and the final aggregation of pixels in a semantic concept, belong to two different epistemic levels (Fonseca and Martin 2007). They are created with different objectives and they have different objects. The aggregated set deals with general assumptions concerning the explanatory invariants of a domain – those that provide a framework enabling understanding and explanation of data across all domains, inviting explanation and understanding. Aggregated sets belong to an ontological level (Fonseca and Martin 2007). Individual pure pixels are related to the consequent dimensions of possible variation among the relevant data of a given domain. There is a natural decomposition of information in terms of two necessary but complementary epistemic functions: identification of an invariant background (aggregated sets) and measurement of the object along dimensions of possible variation (the individual pure pixels).

Fonseca and Martin (2007) suggest the use of objectives and object to understand the difference between basic measurements and their later user as semantic concepts. Adapting their proposition to our case, we can say that regarding the *objectives*, aggregated sets provide explanation and information integration grounded in assumptions about invariant conditions that define the domain of interest. Pure pixels, on the other hand, enable the measurement and classification of the observed facts. Now, regarding the *object*, aggregated sets are based on the real world as understood by humans, on the reality, instead of focusing on what can be represented. The object is the representation of the invariant conditions of the domain of interest – the general, and assumed, categories that are taken to define a domain. Pure pixels are based on the permissible range of variation among the facts that, later, must be brought into relation with those categories.

5. Conclusions

For the Semantic Pixel to be used as a key to reach measurements, in the way of an index, it is necessary to define the kinds of operations that have to be implemented in order to achieve this kind of "transparency". In such a scenario, the user knows and mentions only the Semantic Pixel; then, a computational layer on the background translates the user's conceptualization into the values stored in the images and arrays. Initially, we think much in the way of the interpolation, resampling, segmentation and classification techniques that would be necessary. The difference is that the user would not need to know what is going on underneath the hood to get information adapted to his particular conceptualization. That can be defined as a large computational problem, since it must be efficient, automated, and would require large resources in terms of storage and processing. A good definition on what is required here could be put against what is available (and what can be proposed and implemented) as operations in SciDB,

and would generate a to-do list for enhancing array databases to be used in large remote sensing libraries.

As we see it, the implementation of the Semantic Pixel could be something along the lines of the multidimensional arrays of SciDB (Paul; Stonebraker 2010), in which each cell has values in many different dimensions: measurement dimensions (pixel values, obtained directly from the original image), classification results (e.g. vegetation type, land use type, forested/deforested, crop type), instantaneous values of measurable phenomena or indicators. Moving up conceptually, this could be seen as a series of geofields (Câmara, Thomás et al. 1999) that overlap in space materialized as arrays. In order to make the semantic dimensions more evident, a mapping towards implementation as a set of arrays, to fit SciDB's physical model, would be needed.

Regarding the theoretical aspects, the dichotomy between of the pixel origin, strongly related to the measurement, and its application, aggregated in highly semantic concepts needs to be further studied and developed. We used two concepts, object and objectives to understand this dichotomy and later on propose a way to link pixels to higher semantic concepts.

With regards to implementation much needs to be done. With the Semantic Pixel, we intend users to be able to browse the information contained in the images in two different dimensions, using its semantics and its measurement characteristics. Users with different skill levels will use such a system in different ways, choosing to start with the measurement characteristics or the semantic values to perform their information retrieval, according to their individual preferences.

Acknowledgments

Authors acknowledge the support of CNPq (308678/2012-5, 401822/2013-3), and FAPEMIG (CEX-PPM-00518/13), Brazilian agencies in charge of fostering research initiatives.

References

Câmara, G., U. Freitas, et al. (1994). "A model to cultivate objects and manipulate fields." Proceedings 2nd ACM Workshop on Advances in GIS: 20-28.

Câmara, G., R. Thomás, et al. (1999). Interoperability in Practice: Problems in Semantic conversion from current technology to OpenGIS. Interoperating Geographic Information Systems, Springer: 129-138.

Couclelis, H. (1992). People Manipulate Objects (but Cultivate Fields): Beyond the Raster-Vector Debate in GIS. Theories and Methods of Spatio-Temporal Reasoning in Geographic Space. A. U. Frank, I. Campari and U. Formentini. New York, Springer-Verlag. 639: 65-77.

Ferreira, K. R., G. Câmara, et al. (2014). "An algebra for spatiotemporal data: From observations to events." Transactions in GIS 18(2): 253-269.

Feyerabend, P. K. (1993). Against method. London; New York, Verso.

Fonseca, F. and J. Martin (2007). "Learning the Differences Between Ontologies and Conceptual Schemas Through Ontology-Driven Information Systems." JAIS - Journal

of the Association for Information Systems - Special Issue on Ontologies in the Context of IS 8(2): 129-142.

Fonseca, F. T., C. A. Davis Jr., et al. (2009). "Spatial Data Infrastructures for the Amazon: a First Step towards a Global Forest Information System." Earth Science Informatics 2(4): 188-191.

Kant, I. (1996). Critique of pure reason. Indianapolis, Ind., Hackett Pub. Co.

Koenderink, J. J. (2005). Geometric Framework for Image Processing. Handbook of Geometric Computing. E. B. Corrochano. Berlin, Springer: 171-202.

Kuhn, T. S. (1996). The structure of scientific revolutions. Chicago, IL, University of Chicago Press.

Kuhn, W. (2009). A functional ontology of observation and measurement. International Conference on GeoSpatial Semantics - GeoS, Berlin, Springer.

Lakatos, I. (1970). Falsification and the Methodology of Scientific Research Programmes. Criticism and the growth of knowledge. I. Lakatos and A. Musgrave. Cambridge, England, University Press.

Paul, G. B. Overview of sciDB: large scale array storage, processing and analysis. Proceedings of the 2010 ACM SIGMOD International Conference on Management of data. Indianapolis, Indiana, USA, ACM.

Popper, K. R. (2002). The Logic of Scientific Discovery. London; New York, Routledge.

Sinton, D. (1977). The Inherent Structure of Information as a Constraint to Analysis: Mapped Thematic Data as a Case Study. First International Advanced Study Symposium on Topological Data Structures for Geographic Information Systems, Dedham, MA, Laboratory for Computer Graphics and Spatial Analysis.

Stonebraker, M. (2010). "SQL databases v. NoSQL databases." Communications of the ACM 53(4): 10-11.

Weinberg, J. R. (1960). An Examination of Logical Positivism. Paterson, N.J., Littlefield Adams.

Automated Production of Volunteered Geographic Information from Social Media

Maxwell Guimarães de Oliveira, Cláudio de Souza Baptista, Cláudio E. C. Campelo, José Amilton Moura Acioli Filho and Ana Gabrielle Ramos Falcão

Information Systems Laboratory – Federal University of Campina Grande (UFCG)
Av. Aprígio Veloso, 882, Bloco CN, Bairro Universitário – 58.429-140
Campina Grande – PB – Brazil

maxwell@ufcg.edu.br, {baptista, campelo}@dsc.ufcg.edu.br,
{joseamilton, anagabrielle}@copin.ufcg.edu.br

Abstract. *The easy production of data with geographic context has enabled a deeper engagement with people and has led to the emergence of Location-Based Social Networks - LBSNs. Such environments have proved to be very useful in the context of smart cities, however, one of the main challenges has been how to keep users willing to contribute and keep the LBSNs in a continuous operation. Concerning this problem, we propose an automated production of Volunteered Geographic Information - VGI - based on Geographic Information Retrieval techniques with the aim of providing valuable and up-to-date information for LBSN environments taking advantage of social media messages around the web. A prototype software was developed and evaluated through a case study using microtexts.*

1. Introduction

Volunteered Geographic Information (VGI) has emerged in the last years as an alternative spatial data source on the Web. It consists of data pooled with the geographic context, which is produced and disseminated by individuals spread throughout the world, forming an environment known as Crowdsourcing [Surowiecki 2005]. These individuals are called volunteers and most of these volunteers are not experts in Geography or Geographic Information Sciences, but ordinary people interested in sharing their viewpoints and knowledge about geographic locations [Goodchild 2007].

The easy production of data with geographic context has enabled a deeper engagement with people in everything involving location. It can be explained by the technological evolution of the last years and novel tendencies of the Web 2.0, including the emergence of devices featuring GPS and the spread of Internet connectivity around the world. Users have been increasingly consuming this type of information by means of location-based applications, and also sharing this information in several domains.

This scenario has led to the emergence of applications such as the Location-Based Social Networks (LBSN). The LBSNs provide context-aware services which allow assigning users to content [Vicente et al. 2011], and provide many different types of services, from entertainment to public utilities. In such a social network, much information is voluntarily created, be it textual, multimedia or geographic. Falcão et al. (2012) developed Crowd4City, a LBSN that can be applied to the domain of smart

cities, which supports participatory human sensors, aiming to create an environment for identification and discussion of matters concerning the government of the cities, a common interest of the population.

Despite initiatives like Crowd4City, one of the main challenges in the use of human sensors has been keeping them willing to contribute and consequently maintain the LBSNs in a continuous operation. Only a few users are in charge of providing a significant volume of information. This phenomenon is visible in terms of geographic location, where many areas around the world are mapped by just one user [Haklay and Weber 2008]. One of the factors regarding users' motivation can be associated with the existence of costs for these volunteers. These costs can be inherent to the learning curve for correct operationalization of a LBSN, or related to the contribution routines. These costs can also be associated with the volunteers' available time and demands persistence from them. Therefore, it becomes necessary to find alternatives that will allow keeping the LBSNs up-to-date even when the volume of contributions of the volunteers is below the expected.

One of the purposes of researches in Geographic Information Retrieval (GIR) is the development of techniques for inference of geographic locations associated to text documents. This is an area full of challenges, involving Natural Language Processing (NLP), handling of uncertainty, disambiguation, context identification, among other tasks [Bordogna et al. 2012]. With GIR techniques, it is possible to process and assign geographic locations to text from websites, blogs and social networks, such as the task known as geoparsing [Purves and Jones 2011]. Hence, we raised the hypothesis that, after the identification of their referenced geographic location, texts from the web, such as messages publicly exchanged in social networks, could automatically turn into useful information in applications such as the LBSNs. Thus, their authors can become non-intentional volunteers in the production of the VGI.

Several researches have addressed the assignment of geographic locations on web documents (Georeferencing) [Rupp et al. 2013] [Watanabe et al. 2011], including social network messages. However, the percentage of the information concerning this geographic context is still very low. Furthermore, approaches based on matching the users' locations and their messages have proved to be very inaccurate, since users can freely disseminate information about the most diversified geographic contexts, which, in most times, mismatch their geographic position at the very moment the information is shared.

On that account, this paper presents an approach for the automated production of VGI based on the application of geoparsing and georeferencing techniques to texts published on the web, especially on social media. We have kept our focus in the body of messages and therefore do not considered previously geotagged texts due to such occurrence is still low. Furthermore, we cannot ensure that an embedded geolocation is the same location that message refers to. The VGI produced by the authors of the processed texts will become available for the users of a LBSN. These users will be the main consumers and also validators of this information, being capable of pointing incoherences as well as stressing the relevance of that information for other users of the network, enriching the crowdsourcing environment.

The main contributions of this paper are: the development of an artifact for automatic production of Volunteered Geographic Information, based on the content of social media texts; and a discussion about geoparsing in informal texts published in microblogs and the value that such information may reveal whether the geographic context is explored. The remainder of this paper is structured as follows. Section 2 discusses related work. Section 3 describes our approach. Section 4 addresses a case study carried out to evaluate the proposed ideas. Finally, section 5 concludes the paper and highlights further work to be undertaken.

2. Related Work

Research on VGI has prevailed in several parts of the world. Besides Computer Science, many correlated disciplines, such as Geography [Goodchild 2007], Geographic Information Science [Du et al. 2011] [Haklay et al. 2010] [Jackson et al. 2010] and Human Factors [Parker et al. 2011] have investigated issues concerning this kind of volunteered information.

One of the most representative VGI project is the OpenStreetMap (OSM) [Haklay and Weber 2008] [Koukoletsos et al. 2012]. The OSM database consists of a significant collection of volunteered spatial data based on the Wikipedia collaborative model [Mooney and Corcoran 2012]. The OSM project has received many contributions from the community. Haklay (2010), for instance, has focused on assessing VGI quality and how VGI can be reliable and usable. Ballatore and Bertolotto (2011) focused on semantic relationships within OSM data. They highlight how OSM is spatially rich but semantically poor and investigate ways of linking OSM to other distributed repositories.

Besides the OSM project, several works have revealed VGI as a promising research field. Horita et al. (2013) made a thorough literature review on VGI with the objective of verifying its applicability for aiding in disaster management. In that study, it was possible to observe that the VGI has been more frequently used in fires and floods. Havlik et al. (2013) discussed VGI mobile applications concerning several aspects, such as functionalities and user experience. Ballatore et al. (2013) explored the semantic side of VGI and presented a technique for computing the semantic similarity of geographic terms in VGI based on their lexical definitions and using WordNet. The authors based themselves on the intuition that similar terms tend to be recursively defined by similar terms.

While the research on VGI is still relatively novel, the research on GIR has many studies focused on the identification and indexing of geographic locations through the application of Natural Language Processing (NLP) techniques. Some related works on GIR will be described as follows. Rupp et al. (2013) discussed the customization of geoparsing and georeferencing tools to be applied in collections of historical texts. The authors made an analogy between the storage/indexing of files about the medieval era and the storage/indexing of Twitter feeds, and discussed questions involving standardization and use of gazetteers. There is no discussion about the spatial precision of the geoparsing, but this could be a motivational factor for such customization.

Liu et al. (2013) proposed QGIR, Qualitative Geographic Information Retrieval, as a better option to deal with geographic information described in natural language in web documents. The authors argue the replacement of GIR by QGIR for cases where

the place name and thematic representations are necessary, considering the use of semantic spatial relations and domain-specific ontologies. An experiment was carried out in order to compare QGIR with the standard GIR, and the results proved the superiority of QGIR for queries like "precious metals in the Hebei province". Freire et al. (2011) described an approach for recognition of place names expressed in metadata of digital libraries. That approach should be better at capturing features of the non-structured text found in metadata records and at the exploration of the relevant information in the structured data of those records.

Watanabe et al. (2011) proposed an automatic method for identification of geographic location in non-geotagged tweets. Such method is based on the clustering of messages according to the type of event, considering short time intervals, small geographic areas and geotagged tweets. Thus, geotagged tweets are used to allocate geotags in tweets which do not have the geographic tag yet. The authors do not consider the possibility of the geotagged tweets having a different geographic reference than the location discussed in the messages. Also, it is possible that users are not necessarily talking about their current locations. Therefore, there is a possibility of errors in the geographic precision and this must be considered. In a similar way, Jung (2011) presented a method for analyzing sets of microtexts, aiming at identifying contextual clusters of tweets. By establishing a contextual relation between the messages, a set of microtexts can be considered as a single document and make the process easier for the geoparsers. This task, however, can be very costly, depending on the volume of related tweets. In addition, there is also a possibility of errors in the geographic precision.

Campelo and Baptista (2009) proposed a model for extraction of geographic knowledge from web documents. They developed GeoSEn, a search engine with geographic focus, which enables the geographic indexing of documents extracted from the web. Thus, it is possible to infer geographic locations cited in a text written in Portuguese in a political-division hierarchy, going from the least precise levels (Brazilian regions) to the most precise ones (cities).

As we can notice, there are several researches on identification of geographic location in social media messages focusing exclusively on the text. However, the majority does not address specific issues of Portuguese language. Furthermore, they do not also address LBSN domain and the aim of providing valuable information for such environments in an automated way. In this sense our proposal comes as a solution to cover this gap.

3. Automated Production of VGI based on Crowdsourced Social Media

This section presents our approach for automated production of VGI based on crowdsourced Social Media.

The main objective of this approach is the automated VGI production based on information published on social networks and focusing solely on microtexts. Thus, the expected result is the production of spatiotemporal markers with the content of these messages, which can be widely viewed and handled by the users of a LBSN. This spatiotemporal information may help users interested in learning more about specific geographical locations, for instance, through people who freely share information in social media. An illustration of the proposed approach is presented in Figure 1.

Figure 1. The main idea of our proposal: turning social network messages into spatiotemporal markers in a LBSN

Figure 1 (left side) illustrates the social networks as information sources for the production of the VGI visualized in a LBSN such as Crowd4City (right side), for example. In this context, each message posted by the users of these networks can be turned into a spatiotemporal marker, which can then be used by the users of the LBSN through recommendation and feedback actions, or just getting information. It is important to highlight that the VGI term in our work is related to the spatiotemporal markers that will be produced automatically by social media users who become volunteers even without necessarily access a LBSN.

In order to achieve such goal, it is necessary to have a computational processing involving capture and treatment of information and application of GIR techniques. This processing is illustrated in Figure 2.

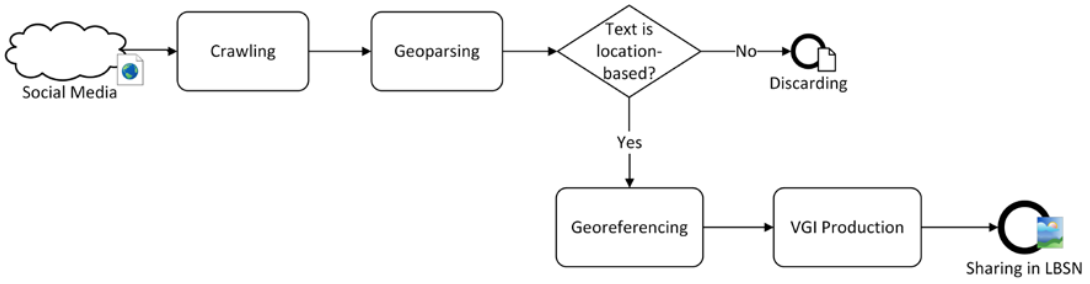


Figure 2. Computational processing flow for automated VGI production

As shown in the flow presented in Figure 2, the computational processing of this approach involves, basically, four distinct stages: Crawling, Geoparsing, Georeferencing and VGI Production. The initial stage is Crawling, in which occurs the capture of the messages posted on the social networks. We developed a real-time algorithm to capture microtexts (tweets) posted on Twitter[1]. This algorithm focuses on the original text of the messages posted on the network, discarding the other metadata of the tweets, except the timestamp containing the time the message was published.

[1] Twitter: http://www.twitter.com/

Once captured in the crawling stage, the microtexts are submitted to the geoparsing stage. In order to accomplish this stage, we used the GeoSEn Geoparser [Campelo and Baptista 2009], which is responsible for the detection of geographic terms in the process of parsing the analyzed texts written in Portuguese. At this stage, all the candidate locations are identified and then sent to the next stage, in which the text will be georeferenced. Figure 3 illustrates a microtext after the geoparsing stage, where the candidate locations are detected and highlighted.

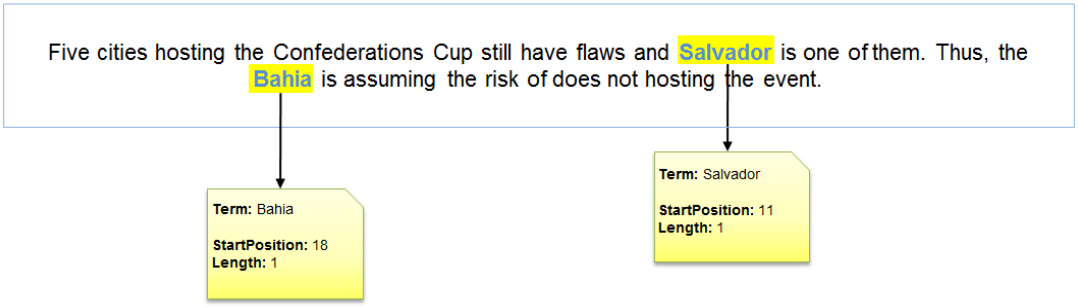


Figure 3. Result of the Geoparsing process applied to microtext (translated from Portuguese)

In Figure 3, it is possible to view all the candidate locations identified in the sample microtext. The Geoparser considers information such as the position of the term in the text and its length, that is, the number of words that form the term. Such position of the term can be used to correlate spatial terms which may appear closely in the messages. In the case where the geoparsing of a microtext returns an empty set of candidate locations, this microtext is discarded and its VGI production process is interrupted.

In the georeferencing stage, the candidate locations pass through a relevance evaluation in order to define the geographic scope of the microtext. In this stage, we used the Geo Scope Modeler featured by GeoSEn. The process of modeling the geographic scope explores the geographic hierarchy

$$\text{city} \rightarrow \text{micro-region} \rightarrow \text{mesoregion} \rightarrow \text{State} \rightarrow \text{region}$$

in order to generate the scope and compute the relevance for its highest levels, based on references found in lower levels. Therefore, the most precise geographic level which can be employed in the production of the VGI from a microtext is the City level. In order to georeference a microtext, the local gazetteer of GeoSEn containing all of the spatial data, structured according to this geographic hierarchy, was also used. The result of the georeferencing stage, applied to the sample microtext of Figure 3, is shown in Figure 4.

In Figure 4 we can notice that only one of the two candidate locations highlighted was considered for the georeferencing of the microtext. Since one of these locations (the city of Salvador) is inside the other one (the State of Bahia), the geographic scope modeling algorithm returned just the most geographically precise.

Finally, the VGI production stage is responsible for producing the spatiotemporal marker that will be shared on the LBSN. The marker is basically formed

by the original microtext captured from the social network, the spatial data obtained in the georeferencing stage and the timestamp of the moment that the message was first published on the social network. For the generation of spatial markers, we compute the centroid points of the geometries georeferenced in the texts. Moreover, these markers produced automatically are assigned an exclusive type defined in the Crowd4City so that they can be easily distinguished from the types originally managed by the LBSN users, like education, transportation, security, etc. Thus, this exclusive type can highlight that the marker was not produced by a LBSN user.

Figure 4. Result of the georeferencing process applied to a microtext (translated from Portuguese)

A software application called *text2vgi* was implemented taking into account the whole flow illustrated in Figure 2, which was detailed throughout this section. The purpose of this application is to validate the proposed approach, confirm our raised hypothesis discussed previously in the introductory section, and identify points which may possibly need further improvements in order to ensure the most spatially-accurate VGI production.

4. Non-intentional VGI from non-geotagged microtexts – A Case Study

In this section, we present a case study using the *text2vgi* software application with microtexts from a social network.

4.1. Methodology

Our study used a dataset formed by 329,732 microtexts written in Portuguese, published on Twitter during the FIFA's Confederations Cup, which took place in Brazil in 2013. We chose to use this dataset because it is related to an event in which people normally use terms that can be associated to geographic location, such as the name of the host cities. The methodology used for conduction of this study is illustrated in Figure 5.

The Crawler implemented in *text2vgi* was responsible for capturing the messages and storing them in a local database. As the messages were received by the application, the geoparser was activated to identify the candidate locations. Then, the

georeferencing module modeled the geographic scope of the microtexts that presented at least one candidate location. Finally, the VGI production module concluded the work creating the spatiotemporal marker.

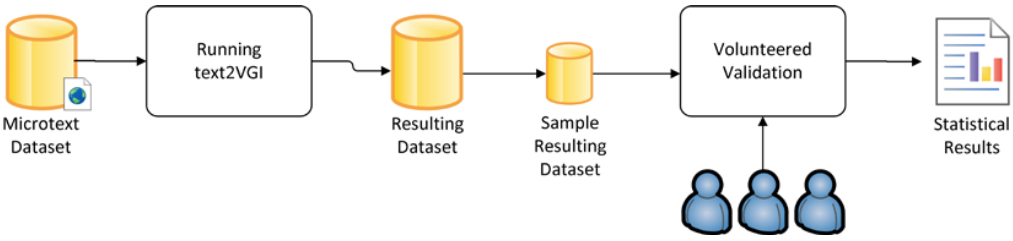


Figure 5. The process flow for the case study

4.2. Volunteered Validation

The whole set of microtexts processed by *text2vgi* needed to be validated concerning the identified geographic locations and the spatiotemporal markers created. Thus, we could measure the performance of the automated production of VGI. For such, we needed to recruit some volunteers and instruct them in the validation process. It was also necessary to develop a web application in order to assist these volunteers on validating the processing performed.

The application for volunteers' assistance presented a random list of processed microtexts, which were to be analyzed individually. For each one of these validated microtexts the volunteers answered the following questions: geoprocessing accuracy (boolean, star ★ [1→5]), if it refers to more than one place (boolean) and if it can be more precise (boolean).

The geoprocessing accuracy question could receive the combination (TRUE, ★★★★★) as answer in the cases in which the georeferencing was totally accurate according to the georeferencing strategy used, or in the cases where the VGI was not produced because the microtext did not express any geographic location. It could also receive the tuples (FALSE, [0 | ★ | ★★ | ★★★ | ★★★★]) as answer, depending on the geographic and semantic distances between the georeferencing location and the location identified manually by the volunteers on reading the microtext. One example of geographic distance is a microtext expressing the city of "Campina Grande" however it was georeferenced as "Paraíba" (the State) or "Nordeste" (the Region). The semantic distance is related to misunderstanding of the georeferencing such as a microtext expressing the "Bahia" (Football Team) in which was georeferenced as "Bahia" (the State) instead of non-location.

The question about whether a microtext refers to more than one place could receive TRUE when the microtext refers to more than one geographic location and, therefore, would allow the production of more than one spatiotemporal marker for the same microtext; and receives FALSE, otherwise. Finally, the question about whether a microtext can be more precise could receive TRUE when the microtext presents evidence that might make the modeling of the geographic scope more precise at city level, such as neighborhood names, streets, or specific buildings, such as parks, squares, stadiums and tourist spots.

From the whole set of processed microtexts, 2.3% (about 7,500) had at least one geographic location automatically assigned by *text2vgi* and could then produce spatiotemporal markers. It is important to highlight that there might be several microtexts which did not express a geographic location and such fact can explain this rate. The volunteered validation may help to understand this aspect.

Considering the huge volume of microtexts of the dataset used in this study, a random sample of these microtexts needed to be defined so it could then be validated by the volunteers. With a trust level of 99% and a sampling error of 0.65%, the sample validated by the volunteers consisted of 35,120 microtexts. In this sample, 975 microtexts (2.7%) had a geographic location automatically assigned by *text2vgi*, nearly the same proportion presented by the whole set of processed microtexts. Since the validation was performed by humans, we still consider a margin of error of 2.0%.

4.3. Results

The mean processing time of each microtext in *text2vgi*, from the moment of the capture of the message to the production of the spatiotemporal marker, was of 0.25 seconds. It took nearly 23 hours to process the whole dataset in only one computer, with an Intel Core i7 processor, 8 GB of RAM and 1 single thread.

Considering the sample validated by the volunteers, Figure 6 presents the results for true positives, when the geographic location was identified correctly; false positives, when the geographic location was not identified correctly; true negatives, when there was no geographic scope assigned due to the lack of evidence in the text; and false negatives, when no geographic scope was assigned, but there was evidence for it.

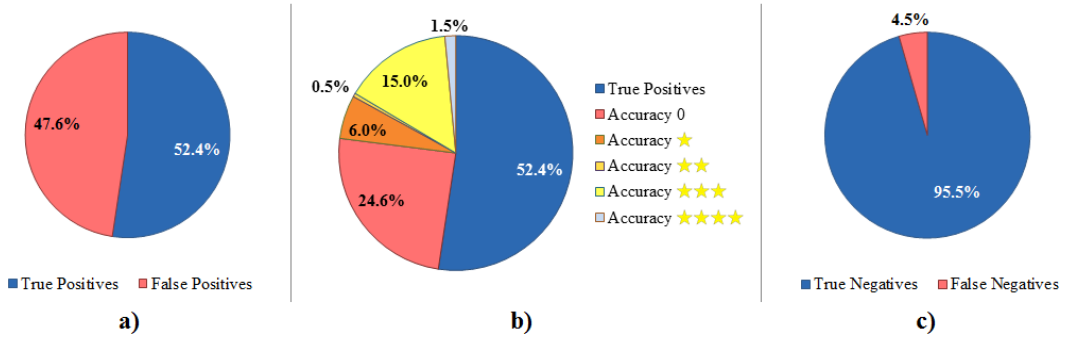


Figure 6. Pie charts representing the percentages of each result: a) True/False Positives Relation, b) True/False Positives Relation considering the False Positives in five subdivisions, and c) True/False Negatives relation

Figure 6a show that there was a balance between true and false positives, if we consider as true positives only the 100% precise location detections. In Figure 6b, it is possible to see the false positives in five classifications levels. Each classification level represents how geographically close the false positive was to a true positive. We can notice the false positives that are very far from the location expressed in the microtext (which received no stars in the accuracy question), represent only 24.6% - about half the total number of false positives. Finally, in Figure 6c, it is possible to observe a good result for true negatives. It confirms the lower rate of the processed microtexts which

had at least one geographic location automatically assigned by *text2vgi*: in fact there were several microtexts that did not express one location at least.

The validation performed by the volunteers on the microtexts also resulted in the following data:

- 16.6 % of the microtexts have evidence for georeferencing of more detailed geographic locations. A georeferecing strategy which takes this aspect into account may improve the overall accuracy;
- 3.2 % of the microtexts have evidence for the inference of more than one geographic location, thus producing more than one spatiotemporal marker.

Table 1. Statistical results of volunteered validation over VGI produced automatically

Overall Accuracy	**Precision**	**Recall**	**F-Measure**
74.1 %	92.3 %	52.6 %	0.67

Table 1 presents four metrics for evaluating the overall performance of VGI produced automatically by *text2vgi* and validated by volunteers during this case study. Among the analyzed metrics, we can notice a low recall rate, that is, 47.4% of the microtexts with geographic location evidence were not correctly identified by *text2vgi*.

Nevertheless, this result was already expected, since the geographic scope considered in the georeferencing strategy used considers only locations related to the Brazilian political territorial division. The geographic references that may be expressed in the set of microtexts used such as soccer stadiums and airports ended up not being properly interpreted. However, it is important to highlight the good precision rate resulted, which is justified by the number of true negatives.

5. Conclusion and Further Work

In this paper, we presented an approach for automated VGI production based on geoparsing and georeferencing of texts published on the web. Such approach was conceived with the objective of turning web authors into volunteers in the VGI context, contributing to the indirect production of information in a Location-based Social Network.

A prototype, called *text2vgi*, was implemented with the goal of validating the ideas proposed by our approach. In order to evaluate the prototype in a real context, we carried out a case study using a set of microtexts in the Portuguese Language concerning a sporting event of large impact on media, the 2013 FIFA's Confederations Cup, held in Brazil.

Overall, the achieved results were considered satisfactory. However, we have confirmed the need for improving the georeferencing strategy in order to increase the amount of VGI produced from microtexts, to improve the spatial accuracy of the spatiotemporal markers created and to achieve better results for the recall and F-Measure. It is important to consider points of interest such as soccer stadiums and airports, and other buildings and well known places in a city context. Thus, the automatically produced VGI will become more spatially precise and the user's experience in the LBSN will be improved.

As future work, we consider the implementation of georeferencing strategies to address the specific treatment of microtexts like informal language. Besides, we will seek the development of heuristics that increase the precision of the locations detected, and consequently improve the F-Measure. Other future direction of our work is to improve our approach for production of VGI based on microtexts in other languages such as English, Spanish and French.

References

Ballatore, A., Wilson, D. C. and Bertolotto, M. (2013) "Computing the semantic similarity of geographic terms using volunteered lexical definitions", International Journal of Geographical Information Science, Taylor & Francis, vol. 27, no. 10, p. 2099-2118.

Ballatore, A. and Betolotto, M. (2011) "Semantically enriching VGI in support of implicit feedback analysis", Web and Wireless Geographical Information Systems, LNCS, vol. 6574, p. 78-93.

Bordogna, G., Ghisalberti, G. and Psaila, G. (2012) "Geographic information retrieval: Modeling uncertainty of user's context", Fuzzy Sets and Systems, vol. 196, p. 105-124.

Campelo, C. E. C. and Baptista, C. S. (2009) "A Model for Geographic Knowledge Extraction on Web Documents", In: Advances in Conceptual Modeling - Challenging Perspectives, LNCS 5833, Edited by Carlos Alberto Heuser and Günther Pernul, Springer Berlin Heidelberg, p. 317-326.

Du, H., Anand, S., Morley, J., Leibovici, D., Hart, G. and Jackson, M. J. (2011) "Developing open source based tools for geospatial integration", In: Proceedings of the 3rd Open Source GIS Conference, Anonymous Nottingham University, UK.

Falcão, A. G. R., Baptista, C. de S. and Menezes, L. C. de (2012) "Crowd4City: Utilizando Sensores Humanos como Fonte de Dados em Cidades Inteligentes" (in Portuguese), In: Proceedings of the 8th Brazilian Symposium on Information Systems, São Paulo, Brazil.

Freire, N., Borbinha, J., Calado, P. and Martins, B. (2011) "A Metadata Geoparsing System for Place Name Recognition and Resolution in Metadata Records", In: Proceedings of the 11th Annual International ACM/IEEE JCDL, Ottawa, Canada, p. 339-348.

Goodchild, M. F. (2007) "Citizens as voluntary sensors: Spatial data infrastructure in the world of Web 2.0", International Journal of Spatial Data Infrastructures Research, vol. 2, p. 24-32.

Haklay, M. (2010) "How good is volunteered geographical information? A comparative study of OpenStreetMap and ordnance survey datasets", Environ Plan B Plan Des, vol. 37, no. 4, p. 682-703.

Haklay, M., Ather, A. and Basiouka, S. (2010) "How many volunteers does it take to map an area well?", In: Proceedings of the GIS Research UK 18th Annual Conference, University College London, UK, p. 193-196.

Haklay, M. and Weber, P. (2008) “OpenStreetMap: user-generated street maps”, IEEE Pervasive Computing, vol. 7, no. 4, p. 12-18.

Havlik, D., Soriano, J., Granell, C., Middleton, S. E., van der Schaaf, H., Berre, A. J. and Pielorz, J. (2013) “Future Internet enablers for VGI applications”, In: EnviroInfo, Edited by B. Page, A. G. Fleischer, J. Göbel and V. Wohlgemuth, Shaker, p. 622-630.

Horita, F. E. A., Degrossi, L. C., Assis, L. F. G. de, Zipf, A., Albuquerque, J. P. (2013) "The use of Volunteered Geographic Information (VGI) and Crowdsourcing in Disaster Management: a Systematic Literature Review", In: Proceedings of 19th AMCIS, Chicago, USA, p. 1-10.

Jackson, M. J., Rahemtulla, H. and Morley, J. (2010) “The Synergistic use of authenticated and crowd-sourced data for Emergency response”, In: Proceedings of the 2nd Int. Workshop on Validation of Geo-Information Products for Crisis Management (VALGEO), European Commission Joint Research Centre, Ispra, Italy, p. 91-99.

Jung, J. J. (2011) “Towards named entity recognition method for microtexts in online social networks: a case study of Twitter”, In: Proceedings of the International Conference on Advances in Social Network Analysis and Mining, p. 563-564.

Koukoletsos, T., Haklay, M. and Ellul, C. (2012) “Assessing Data Completeness of VGI through an Automated Matching Procedure for Linear Data”, Transactions in GIS, vol. 16, no. 4, p. 477-498.

Liu, L., Gao, Y., Lin, X., Guo, X. and Li, H. (2013) “A framework and implementation for qualitative geographic information retrieval”, In: Proceedings of 21st International Conference GeoInformatics, p. 1-4.

Mooney, P. and Corcoran, P. (2012) “The Annotation Process in OpenStreetMap”, Transactions in GIS, vol. 16, no. 4, p. 561-579.

Parker, C. J., May, A. J. and Mitchell, V. (2011) “Relevance of volunteered geographic information in a real world context”, In: Proceedings of GIS Research UK 19th Annual Conference, University of Portsmouth, UK.

Purves, R. and Jones, C. B. (2011) “Geographic Information Retrieval”, SIGSPATIAL Special, vol. 3, no. 2, p. 2-4.

Rupp, C. J., Rayson, P., Baron, A., Donaldson, C., Gregory, I., Hardie, A. and Murrieta-Flores, P. (2013) “Customising Geoparsing and Georeferencing for Historical Texts”, In: Proceedings of the International Conference on Big Data, IEEE, p. 59-62.

Surowiecki, J. (2005), The Wisdom of Crowds, Anchor.

Vicente, C. R., Freni, D., Bettini, C. and Jensen, C. S. (2011) “Location-Related Privacy in Geo-Social Networks”, IEEE Internet Computing, vol. 15, no. 3, p. 20-27.

Watanabe, K., Ochi, M., Okabe, M. and Onai, R. (2011) “Jasmine: a real-time local-event detection system based on geolocation information propagated to microblogs”, In: Proceedings of the CIKM’11, Glasgow, Scotland, p. 2541-2544.

A Comparative Analysis of Development Environments for Voluntary Geographical Information Web Systems

Jean H. S. Câmara, Thales T. Almeida, Denis R. Carvalho, Thiago B. Ferreira, Allan F. Balardino, Gilberto V. Oliveira, Fabio J. B. Fonseca, Ricardo S. Ramos, Wagner D. Souza, Jugurta Lisboa-Filho

Departamento de Informática – Universidade Federal de Viçosa (UFV)
36570-900 – Viçosa – MG – Brasil

jeanhsc2010@gmail.com, thalesalmeida88@yahoo.com.br, denis.carvalho@ifmg.edu.br, thiagao.ti@gmail.com, allan.balardino@ufv.br, gilbertooliveiralf@gmail.com, fabio.fonseca@ufv.br, rsramos07@gmail.com, wagnerdiasdesouza@gmail.com, jugurta@ufv.br

Abstract. *As mobile devices, access to geographical information, and migration from web 1.0 to web 2.0 advance, users started playing the role of consumers, producers, and communicators. As a result, several internet systems have spawned that collect Volunteered Geographical Information (VGI). VGI collection systems often need to be developed within short timeframes. This paper presents a comparative analysis between two environments for VGI-system development: Ushahidi Platform and ClickOnMap. This comparison employed a model based on system-quality norms ISO 9126. The results of this comparison may help VGI system developers choose the tool with the most appropriate characteristics to the goal intended when creating the system.*

1. Introduction

The technological and conceptual evolution of the Web in the 1990s and 2000s brought a new outlook to its users. In the current Web format, users may play the role of consumers, producers, and communicators within this environment. Thus, they become largely responsible for the creation, storage, and dissemination of information [Carvalho 2008]. In this context, the concept of user-generated content (UGC) arises in the contemporary scenario, in which sites such as Flickr[1] and YouTube[2] every day receive many contributions and shares. Likewise, other social media systems thrive and take up a key role in publicizing worldwide events with information being shared nearly in real time by users, such as through Facebook[3] or Twitter[4] [Amaral 2012].

During the consolidation of new Web, a specific type of UGC arose, in which the data involved have a spatial component for location and shape. Goodchild (2007) termed this phenomenon Volunteered Geographic Information (VGI), which involves associating concepts of Neogeography [Turner 2006], Collective Intelligence [Lévy

[1] https://www.flickr.com/
[2] https://www.youtube.com/
[3] https://www.facebook.com/
[4] https://twitter.com/

1999], and Web 2.0 resources [O'reilly 2007]. In this type of system, citizens are voluntarily and collaboratively used as human remote sensors, collecting data from events and making them available through Geobrowsers [Goodchild 2007].

User participation in acquiring the data that feed such systems is closely related to the popularization and convenience of using GPS (Global Positioning System) in several new forms of technologies such as smartphones. These tools allow the user to participate in the production of geographic information in an easy way. These activities were previously restricted to technicians specializing in Geographic Information Systems, typically linked to governmental organizations and private businesses that provide geospatial data [Silva 2008].

Collaborative data acquisition is advantageous in several ways, such as cutting down the cost and time of data production. This gain in dada collection may be crucial in several situations, mainly in emergencies such as natural disasters, since collecting and distributing data nearly instantaneously may be a requirement [Georgiadou 2011]. If no VGI system is available that meets the needs of the population at that moment, a framework should be used that enables easy and quick creation and customization of such a system [Souza et al. 2014].

This paper aims to present, discuss, and compare the construction and functionality of two distinct frameworks, namely Ushahidi Platform [Okolloh 2009] and ClickOnMap [Souza et al. 2014]. The experiments were developed within the Spatial Databases course of Graduate Program on Computer Science at Federal University of Viçosa. The remaining of the paper is structured as follows. Section 2 describes the tools used in the VGI system development process. Section 3 reports on the systems developed. Section 4 describes the comparison model applied, besides approaching the results obtained. Section 5 presents the study's conclusions.

2. Environments for Collaborative System Development

A collaborative system often needs to be developed within a short timeframe. In order to achieve this, some frameworks exist that speed up its creation, such as ClickOnMap, Ushahidi Platform, and Davis's framework. ClickOnMap is a framework that enables creating and customizing a collaborative Web system quickly and intuitively [Souza et al. 2014]. Ushahidi Platform is another framework developed from a site called Ushahidi. This site aimed to map violent events in Kenia in 2008 [Okolloh 2009]. Davis's framework is intended for the creation of VGI applications able to receive contributions via the Web and mobile devices [Davis Jr, et al. 2013].

After these environments were analyzed, Ushahidi Platform and ClickOnMap were compared within each to verify the quality of both collaborative system development environments. These tools were chosen for being freely available on the Internet and for having good supplementary material to aid in installation, which minimizes difficulties and the time spent configuring a new collaborative Web system. At the time of the comparison, Davis's framework was not available to be used. Below, the two frameworks compared are described.

2.1. Ushahidi Platform

This platform can be freely obtained at the company's website[5]. It is quick and intuitive to install, requiring only the administrator's name and e-mail address, the system's name, slogan, and database information. This environment provides one pre-configured system. In order to change the system's configurations, the administrator must access the management panel, shown in Figure 1.

This panel allows the name, slogan, banner, e-mail, language, and other pieces of system information to be changed. In addition, the time zone, default map location, zoom level, and map provider can be changed. The system is able to use the four main base map services, from ESRI, Google, Bing, and OpenStreetMap. This environment also enables creating, editing, and excluding categories, subcategories, users, and contributions. Several plugins are available that enhance the system's functionality.

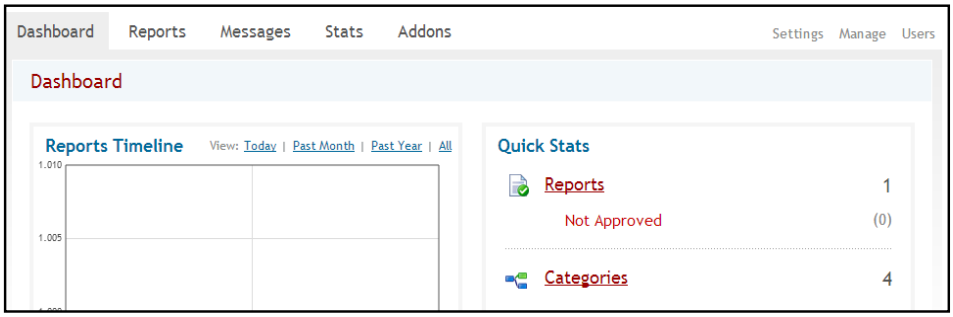


Figure 1. Management panel of Ushahidi Platform

Ushahidi Platform enables creating an online collaborative system with no need for hiring a hosting service. The tool that provides this feature is called Crowdmap. With it, administrators only need to make an account to create their own collaborative system. However, the system created on Crowdmap will be linked to the company's domain. This tool does not provide access to the system's source code.

The systems created using this platform enable point, line, or polygon collaborations that may be filtered according to the categories and subcategories on record. Both registered and anonymous users may collaborate, therefore the administrator must evaluate and approve all contributions. Besides these functionalities, users may comment on a given contribution and evaluate on its reliability. A mobile application is available.

2.2. ClickOnMap

This framework is laid out so that a programmer can, quickly and in few steps, customize a system and begin using its features [Souza et al. 2014]. Actually, ClickOnMap does not require advanced knowledge with programming language, since it has a simple and intuitive customization interface, as shown in Figure 2.

[5] http://www.ushahidi.com/

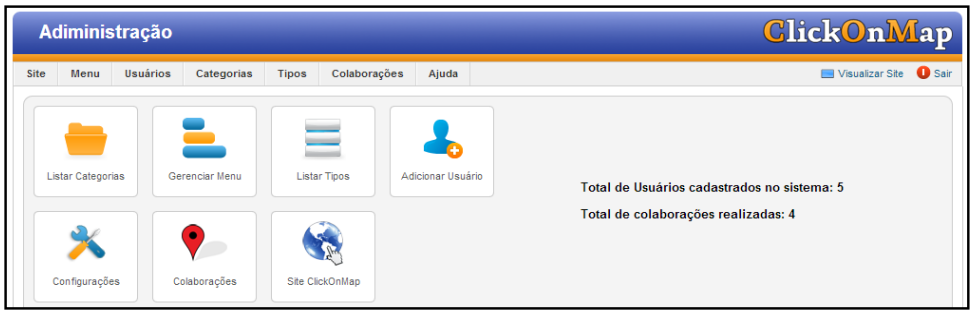


Figure 2. Management panel of ClickOnMap

In the management panel, the system administrator can manage settings, users, and contributions. It enables changing the site's name, system login options, type of map used, and other features. Moreover, categories and types (subcategories), users, and contributions can be added, edited, or excluded [Souza et al. 2014]. ClickOnMap has some features to guarantee and validate VGI quality, namely: VGI score, in which each user rates the VGI between 0 and 5; and user score, through which each action generates a positive or negative score to that user. This score is used for a User Rank, divided into point-range classes. All system users are Moderators, i.e., they can edit any VGI, while Administrators can validate a VGI [Souza et al. 2014].

In addition, the systems created with ClickOnMap have VGI filters and statistics that are shown to the users as graphs, which help in decision-making processes. The system has tools for dynamically analyzing the data of a region. ClickOnMap uses template Dynamic Metadata for VGI (DM4VGI) with dynamic metadata to document and validate VGI quality [Souza et al. 2013]. Therefore, this framework standardizes and facilitates VGI documentation, making searching and accessing a VGI more efficient. DM4VGI also has elements to capture statistical data on VGI use, besides of data about VGI quality.

3. VGI Systems Developed for the Comparison Between the Frameworks

In order to compare the Ushahidi Platform and ClickOnMap frameworks, two different systems were implemented in each of these platforms. Each implementation was carried out by two programmer students. The subjects chosen for these systems were advertising parties around the city and identifying accessibility features/difficulties in public areas. Both systems were applied to Viçosa city, MG, Brazil. Based on the implementation of these systems it was possible discover the functionality of frameworks, allowing to make a comparison between the Ushahidi Platform and the ClickOnMap.

The first system allows its users to inform the place and date of a party. This subject was chosen since college-oriented cities hold many parties and advertising them is key to their success. Using a system to this end makes the advertising process both quicker and cheaper. The second system collects information regarding accessibility in public areas by people with some type of special physical needs. This subject was chosen given the large number of complaints about poor accessibility found in social networks. Moreover, both systems are helpful to the general public and receive its attention for being dynamic [Hirata et al. 2013]. Next, the implementations of the two systems in both frameworks are described.

3.1. Party Advertising - ClickOnMap

The system implemented with ClickOnMap, called *Roteiro de Festas*[6] (Party Guide, in free translation), offers four login options: Using an account on the system, a Facebook or Google+[7] account, or anonymously. Besides this login setting, other settings were also changed: the system's name, contact e-mail address, central coordinates of the area of interest, initial zoom level, and type of map used. Three categories were created: Free Parties, Paid Parties, and College Parties. For each category, a few subcategories were created. Figure 3 shows, on the left, the system's homepage and, on the right, the options for the user to filter contributions according to category and VGI type.

In order to post a contribution, the user must log into the system using one of the four methods mentioned above. In case the user chooses to post anonymously, any contribution they make will need to be approved by the administrator before it is visible to all users. However, it the user is not anonymous, the contribution will be instantly available to all users. Contributions approved by the administrator have icons in a different color than the ones yet to be approved. This makes it easier to differentiate each contribution in the map.

3.2. Party Advertising - Ushahidi Platform

This system was developed using Ushahidi Platform's Crowdmap tool. Thus, a server did not need to be hired since the platform itself offers such service. Nevertheless, the service has a few limitations, e.g., the application source code cannot be directly accessed, which keeps some system features from being customized.

Some initial features were changed, such as the site's name (Party In The Map[8]) and slogan, default language, time zone, e-mail address, and page header image. For the complete customization of the environment, the map information such as default location, map provider, and zoom level were edited, and ten VGI categories were created for music: "Brazilian popular music (MPB)", "Samba/Pagode", "Sertanejo", "Forró", "Rock", "Electronic", "Gospel", "Axé", "Funk", and "Country". An icon was created to represent each category. Figure 4 illustrates the system's home page.

The system allows a point, line, or polygon report to be sent with no need for registration. This way, the reports are not shown in the system until approved by the administrator. When approved, these reports can be visualized in a list or on the map. Besides the settings above, a plugin was installed to enable the map to be visualized full screen.

[6] http://www.ide.ufv.br:8008/roteirodefestas/
[7] https://plus.google.com
[8] https://partinthemap.crowdmap.com/

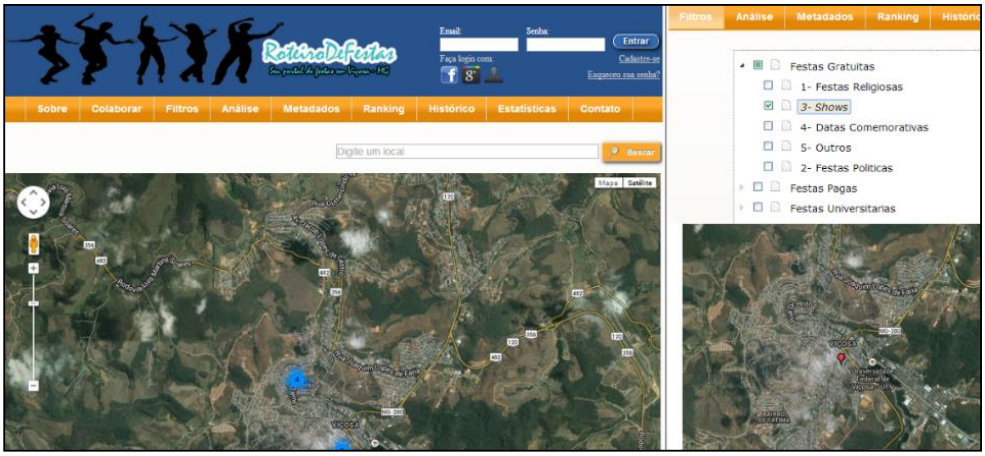


Figure 3. Homepage and filter of the *Roteiro de Festas* system

Figure 4. Party system developed with Ushahidi Platform

3.3 Urban Accessibility - ClickOnMap

A new collaborative environment was created by installing the ClickOnMap framework in a server. The customizations consisted in changing the system's name to *Mão na Roda Viçosa*[9] (Come in Handy Viçosa, in free translation), the contact e-mail, and the visualization as a map for the city of Viçosa. In addition, the same login options as those in section 3.1 were enabled.

The following categories were created: "Buildings", "Urban spaces", and "Furniture and urban equipment". The types related to these categories were registered. Figure 5 illustrates, on the left, the system's home page and, on the right, the window that collects collaborative information such as the title, description, category, type, image, file, video, etc.

[9] http://www.ide.ufv.br:8008/acessibilidade/

3.4 Urban Accessibility - Ushahidi Platform

This system also used the Crowdmap tool of Ushahidi Platform given the upsides already described in previous sections. The following changes were made: name (*Acessibilidade Viçosa*[10], or Accessibility in Viçosa in free translation), default language, time zone, and contact e-mail, among others.

Four categories were added: Great accessibility, Regular accessibility, Poor accessibility, and Reliable contributions. The new contribution alert e-mail feature was enabled in the system. The mobile device application was available for download on the system's page so as to allow contributions to be sent on the go. Figure 6 shows the system's home page.

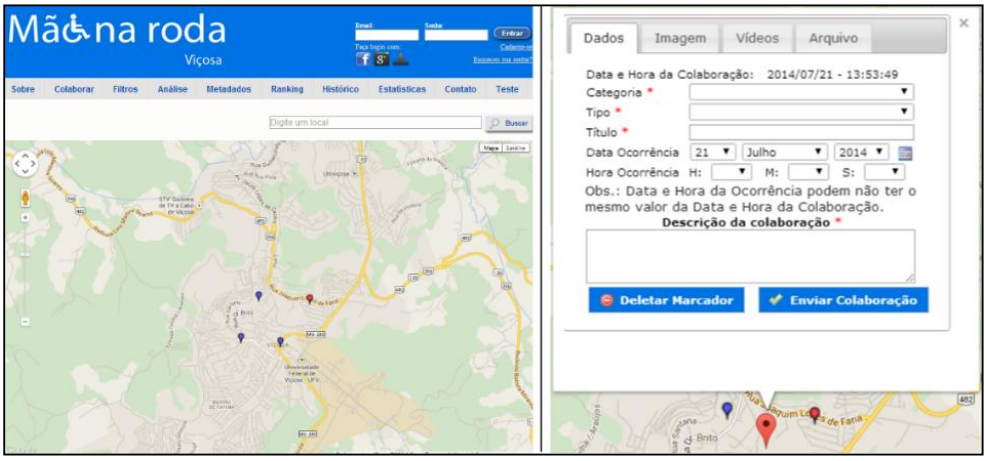


Figure 5. Home page and collaboration form of the *Mão na Roda Viçosa* system

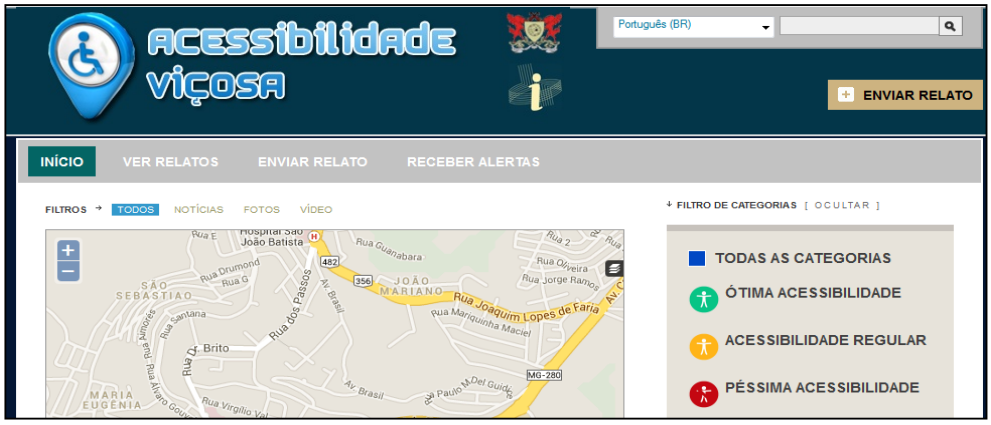


Figure 6. Home page of the *Acessibilidade Viçosa* system

[10] https://acessibilidadevicosa.crowdmap.com/

4. Analysis of Results

The results were assessed based on the evaluation model ISO/IEC/9126-1, used to assess software quality, with a few changes. Therefore, the requirements and attributes needed for a VGI environment were identified and each requirement received a specific weight. This weight is related to its importance class in the system. The levels at which the systems met the requirements were defined. Furthermore, a table was created to assess and compare Ushahidi Platform with ClickOnMap.

4.1. Assessment Model

When evaluating software, well-defined models must be used, and ISO/IEC/9126-1 [NBR ISO/IEC 9126-1 2003] is used as a reference in software quality assessment. However, a Web system must be evaluated differently from a traditional system. The characteristics, the production and execution environments are not the same. Hence, since ISO 9126 is generic, it had to be customized.

According to Moraes (2003), a Web system is a complex environment and evaluating it is a tough task. Thus, Pressman (2001) considers that five of the six features of ISO 9126 apply to Web systems: Usability, Functionality, Reliability, Efficiency, and Maintenance. Table 1 shows the characteristics according to the goals of the VGI systems that fit the generic context presented by ISO 9126 and by Presman (2001).

4.2. Evaluation System

The requirements and attributes needed for a VGI environment were listed. Moreover, two systems were developed using each of the platforms assessed. This enables the results to be analyzed. The platforms were analyzed based on the document that assessed Web system quality since VGI systems fall into this category. However, some attributes particular to collaborative system development environment were added.

Once all attributes and specific characteristics of a VGI system were listed, each attribute must be assigned a weight regarding its importance class in the system. Thus, the attributes were classified as “Essential,” “Important,” or “Desirable,” with weights of 3, 2, and 1, respectively. Besides the weight, it is also important to set the level at which the systems meet the needs of each attribute. These levels may be “Satisfactory” (S), “Partially Satisfactory” (P), or “Non-satisfactory” (N), scoring 2, 1, and 0, respectively.

The software quality evaluation documentation of ISO 9126 (particularly Web systems) has generic characteristics inherent to all systems in this category. Based on this document, all particular attributes were related to the voluntary collaboration systems. Hence, these attributes were evaluated so that the experimental results were analyzed.

4.3. Results

VGI system quality can be measured according to characteristics and functionalities, among which the possibility of anonymous contributions, integration with social networks, friendly interface, and easy installation stand out. Based on the use of the Ushahidi Platform and ClickOnMap environments to develop a VGI system, all key attributes could be related to a VGI system. These attributes were assessed and classified in both systems according to the criteria set previously. Thus, a table (Table 1) was created

to evaluate and compare the two frameworks. Classification (X) is given by the sum of the products of importance weigh (P) and fulfillment level (E) of each attribute (n), represented by formula (1):

$$X = \sum_{i=1}^{n} E_i . P_i \qquad (1)$$

Regarding Usability, it can be seen that none of the systems analyzed is able to meet the requirements of users with some type of physical disability. Therefore, frameworks that support contributions from people with or without physical disabilities may the subject of further research. ClickOnMap has a feedback system so that the user who created the VGI can be informed, via e-mail, about the status of the contribution after it is analyzed by the administrator.

Regarding Functionality, Ushahidi Platform allows for point, line, and polygon contributions, besides allowing collaborations via administrative tool and supporting several map servers. Creating a collaborative system is faster using Ushahidi's Crowdmap since the server does not need to be configured, however, there are fewer features and customizations. Ushahidi Platform also supports several languages and has an application for mobile devices. This facilitates contributions from regions such as rural areas and locations with no wired internet.

ClickOnMap has more types of visualizations than Ushahidi Platform, besides having dynamic data analysis tools. The VGIs could be visualized using markers, kernel maps, or informative clusters. Any user could edit the VGIs (Wiki-Review) and a change log was kept. The dynamic data and statistics analysis tools used pie charts, which were nearly instantaneously redrawn whenever the user visualized another region. It was also possible to analyze data on a specific set of categories and types of a region of interest.

VGI acquisition and documentation in this environment was standardized, thus the data and metadata are interoperable, i.e., data can be collected from different environments that use ClickOnMap and DM4VGI and, for instance, be overlapped for analyzes and information gathering. The databases can also be seamlessly searched and accessed. Another advantage was the possibility of searching the metadata textually, temporally, spatially, and/or thematically, which allowed data to be recovered more efficiently than in Ushahidi Platform.

The ranking system can be an interesting tool to motivate the users to collaborate more and better since they will stand out in the system by making more high-quality contributions. This may lead the users to keep contributing so as to rank even higher or at least to maintain the rank, which leads to a sort of positive competition. This technique to bring users and software closer is widely used in games and satisfactorily reaches its goals. The VGI scores are weighed regarding the user's rank, i.e., users with a better rank position have more weight in the calculation of the final VGI score.

Table 1: Evaluation result

Characteristics	Description	Importance	Ushahidi	ClickOnMap
Usability	Instruction messages	Important	S	S
	Accessibility attributes (e.g., font size)	Important	**N**	**N**
	Help menu	Important	S	S
	Contribution feedback by e-mail	Desirable	N	S
Functionality	Simple and intuitive installation	Important	S	S
	Minimizes development time	Essential	S	S
	Has contribution categories and subcategories	Essential	S	S
	Navigation resources	Important	S	S
	User registration	Essential	S	S
	Information recovery	Important	P	**S**
	Open-source platform	Important	S	S
	Permission levels	Essential	S	S
	Allows anonymous contributions	Essential	S	S
	Supports map servers	Important	S	**P**
	Supports metadata	Important	**N**	S
	Social network integration	Important	S	S
	Mobile device application	Important	S	**N**
	Supports multiple languages	Important	S	**N**
	Shows statistics to users	Essential	**N**	S
	Dynamic content analysis	Desirable	**N**	S
	User score and ranking	Important	**N**	S
	Contribution evaluation	Essential	S	S
	Allows exporting/importing contributions	Important	S	S
	Contribution deletion	Important	S	S
	Contributions created on the management panel	Desirable	S	**N**
	E-mail sent at each contribution	Essential	S	S
	Contribution as different geometric shapes	Important	S	**P**
	Wiki review	Important	**N**	S
Reliability	Intuitive error messages	Important	S	S
	User-validation mechanism	Essential	S	S
	Access credential creation	Important	S	S
	Content-filtering system	Important	S	S
Efficiency	Good user interaction response	Essential	S	S
Maintenance	Supports installing extensions	Important	S	S
	Customization mechanisms	Important	S	S
Total (in points):			126	136

Regarding Reliability, Efficiency, and Maintenance, both systems behaved similarly and had no advantages or disadvantages between each other. Overall, both systems met the goals of a framework intended to easily and quickly create VGI environments. Moreover, they have interesting features to facilitate collaboration, information recovery, and future data analysis.

5. Conclusions

This paper compared two collaborative Web system development environments, Ushahidi Platform and ClickOnMap. The comparison was carried out in a qualitative way, by developing two different systems using each of these platforms, implemented by four groups of programmers. These systems used the subjects of party advertising and accessibility features/difficulties. In face of the results obtained, it can be seen that both Ushahidi Platform and ClickOnMap frameworks have relatively close maturity levels, especially regarding Usability, Reliability, Efficiency, and Maintenance, with a slight advantage for ClickOnMap concerning Functionality.

However, it must be pointed out that each framework has its perks. Ushahidi Platform has a mobile application, supports several languages and different map servers, and allows contributions as points, lines or polygons. ClickOnMap, by its turn, features wiki review, metadata support, dynamic map content analysis, user score and rank, contribution score weighed by user rank, and more efficient information recovery since it allows for textual, temporal, thematic, and spatial searches on standardized metadata.

Acknowledgements

This project was partially funded by CNPq, Fapemig, and CAPES.

6. References

Amaral, I. (2012) “Participação em rede: do utilizador ao ‘consumidor 2.0’ e ao ‘prosumer’”. *CECS/Universidade do Minho e Instituto Superior Miguel Torga*. Braga, Portugal Comunicação e Sociedade, vol. 22, pp. 131 – 147

Carvalho, A. A. A. (2008) "Manual de ferramentas da Web 2.0 para professores”. *Lisboa: DGIDC, Ministério da Educação*. [ISBN 978-972-742-294-4].

Silva, J. C. T., & Davis Jr, C. A. (2008) “Um framework para coleta e filtragem de dados geográficos fornecidos voluntariamente”. *In: IX Brazilian Symposium on GeoInformatics (GeoInfo).* p. 139-144.

Davis Jr, et al. (2013) “A Framework for Web and Mobile Volunteered Geographic Information Applications”. *In: XIV Brazilian Symposium on GeoInformatics (GeoInfo).* p. 147-157.

Georgiadou, Y., et al. (2011) “Sensors, empowerment, and accountability: a Digital Earth view from East Africa”, *Int. Journal of Digital Earth*, 4(4), 285-304.

Goodchild, M. F. (2007) “Citizens as voluntary sensors: spatial data infrastructure in the World of Web 2.0”, *Int. Journal of SDI Research*, v.2, p. 24-32.

Hirata, E., et al. (2013) “Mapeamento dinâmico e colaborativo de alagamentos na cidade de São Paulo”. *Bol. Ciênc. Geod*., sec. Artigos, v. 19, n. 4, p. 602-623.

Lévy, P., & Bonomo, R. (1999) "Collective intelligence: mankind's emerging World in cyberspace", *Perseus Publishing*.

Moraes, E. (2003) "Avaliação de qualidade de aplicações Web". *Projeto final de Curso de Graduação, do DICC/UERJ*.

NBR ISO/IEC 9126-1. (2003) "Engenharia de Software – Qualidade de Produto - Modelo de Qualidade", *Associação Brasileira de Normas Técnicas*.

Okolloh, O. (2009). "Ushahidi, or'testimony': Web 2.0 tools for crowdsourcing crisis information". *Participatory learning and action*, 59(1), 65-70.

O'reilly, T. (2007) "What is Web 2.0: Design patterns and business models for the next generation of software", *Communications & strategies*, 1(17).

Souza, W. D., Lisboa-Filho, J., Camara, J. H. S., Vidal Filho, J. N. (2014) "ClickOnMap: a framework to develop volunteered geographic information systems with dynamic metadata". *In: ICCSA-CTP*, Guimarães, Portugal. Proceedings. Berlin Heidelberg: Springer-Verlag LNCS. p. 1-15.

Souza, W. D., Lisboa Filho, J., Vidal Filho, J. N., Câmara, J. H. S. (2013) "DM4VGI: A template with dynamic metadata for documenting and validating the quality of Volunteered Geographic Information". *In: XIV Brazilian Symposium on GeoInformatics (GeoInfo)*. p. 1-12.

Pressman, R. S. (2001) "Software Engineering: A Practitioner's Approach", *McGraw Hill*.

Turner, A. (2006) "Introduction to neogeography". *O'Reilly Media*, Inc.<http://brainoff.com/iac2009/IntroductionToNeogeography.pdf>. may 2014.

A Hybrid Architecture for Mobile Geographical Data Acquisition and Validation Systems

Claudio Henrique Bogossian[1], Karine Reis Ferreira [1], Antônio Miguel Vieira Monteiro[1], Lúbia Vinhas[1]

[1]DPI – Instituto Nacional de Pesquisas Espaciais (INPE)

{bogo, karine, miguel, lubia}@dpi.inpe.br

Abstract. *Mobile devices, such as smartphones and tablets, are useful tools for in situ gathering information about spatial locations. Specialists need mobile geographical data acquisition and validation systems to be used in fieldwork and in places where there is limited or any network connectivity available. This paper presents an ongoing work on designing and implementing a hybrid architecture for this kind of systems able to work online as well as offline. The offline module of the proposed architecture is based on the OGC Geopackage standard. As part of this work, we tested and evaluated Geopackage documents as interoperable files between spatial data infrastructures and mobile geographical data acquisition and validation systems.*

1. Introduction

The recent advancements of GPS, wireless communication network and portable technologies have motivated the use of mobile devices for *in situ* gathering information about spatial locations and validating geographical data. Tsou (2004) defines the term *mobile GIS* to refer to an integrated technological framework for accessing geospatial data and location-based services through mobile devices, such as smartphones and tablets. He argues that there are two major application areas of mobile GIS, *field-based GIS* and *location-based services*. This work focuses on mobile field-based GIS, that is, mobile systems for geographical data collection and validation in the field.

Two examples of projects that need mobile field-based GIS are PRODES (Monitoring of Brazilian Amazon Rainforest) and DETER (Real Time Deforestation Detection System), developed by INPE [INPE 2014]. PRODES has been yearly monitoring deforestation since 1988 whereas DETER has been producing near real-time deforestation and forest degradation alerts for more than 5 million Km^2 in the Brazilian Legal Amazon. Specialists of these two projects require mobile systems to collect extra information about deforested regions (e.g. photos) and validate them in the field, including places where there is limited or any network connectivity available. Therefore, an essential feature of geographical data collection and validation mobile systems is the capability of working offline.

To meet this demand, this paper presents an ongoing work on designing and implementing a hybrid architecture for this kind of systems able to work *online* as well as *offline*. The offline module of the proposed architecture is based on the Open Geospatial Consortium (OGC) Geopackage standard [OGC 2014]. This work presents an evaluation of Geopackage documents as interoperable files between Spatial Data Infrastructures (SDI) and mobile geographical data acquisition and validation systems.

SDI is a sharing platform that facilitates the access and integration of multi-source spatial data in a holistic framework with a number of technological components including policies and standards [Rajabifard et al 2002] [Mohammadi 2008].

1.1. Related Work

Nowadays, mobile GISs have been widely used in different application areas, including location-based systems [Raper et al 2008], volunteered GIS applications (VGI) [Davis et al 2013] and field-based geographical data acquisition and validation [Tsou 2004] [Poorazizi et al 2008]. This work focuses on this last application area.

Tsou (2004) proposes a generic architecture for mobile GIS where there is a module, called "Geodata cache", responsible for storing geospatial data in a cache located in the mobile storage space or a flash memory card. The idea is to download customized datasets and synchronize them from GIS content servers. Poorazizi et al (2008) proposes two mobile GIS architectures for field geospatial data acquisition: one to work offline (Stand-Alone Client Architecture) and another online (Distributed Architecture).

Differently, we propose a hybrid architecture with two modules for geospatial data access, *online* and *offline*, based on OGC standards (Web Services and Geopackage). The compliance with OGC specifications assures spatial data interoperability between existing SDIs and the mobile systems. Nowadays, many data providers throughout the world have created their own SDIs, organizing and disseminating their geospatial data sets and metadata on the Internet via OGC web services. Accessing spatial data sets from distinct SDIs can improve the geographical data collection and validation task. For example, specialists from INPE's PRODES and DETER projects can use the protected areas disseminated by IBAMA via web services to help with deforested areas validation.

2. The Hybrid Architecture

Figure 1 presents the proposed architecture showing the two modules for accessing geographical data, "Online Data Access" and "Offline Data Access". The "Online Data Access" module accesses geographical data from SDIs through two kinds of well-known OGC web services, Web Map Server (WMS) and Web Feature Server (WFS) [OGC 2006] [OGC 2010]. This module only works online and will be used when there is network connectivity available in the field.

WMS standard provides a simple HTTP interface for requesting geo-registered map images from one or more distributed geospatial databases. The response to the request is one or more geo-registered map images (returned as JPEG, PNG, and others) that can be displayed in a browser application. WFS document specifies the behavior of a service that provides transactions on and access to geographic features in a manner independent of the underlying data store. It specifies discovery operations, query operations, locking operations, transaction operations and operations to manage stored parameterized query expressions.

The "Offline Data Access" module works offline and is responsible for accessing geographical data in the mobile storage memory. We propose to store them in OGC Geopackage files [OGC 2014]. The Geopackage specification defines a SQL database schema designed for the SQLite software library. This schema contains a set of

pre-defined tables with integrity assertions, format limitations and content constraints to store spatial data sets and their metadata. GeoPackage files are platform-independent SQLite database files that contain vector and tiled raster data sets as well as their metadata. They are interoperable across different platforms, including personal computing environments and mobile devices.

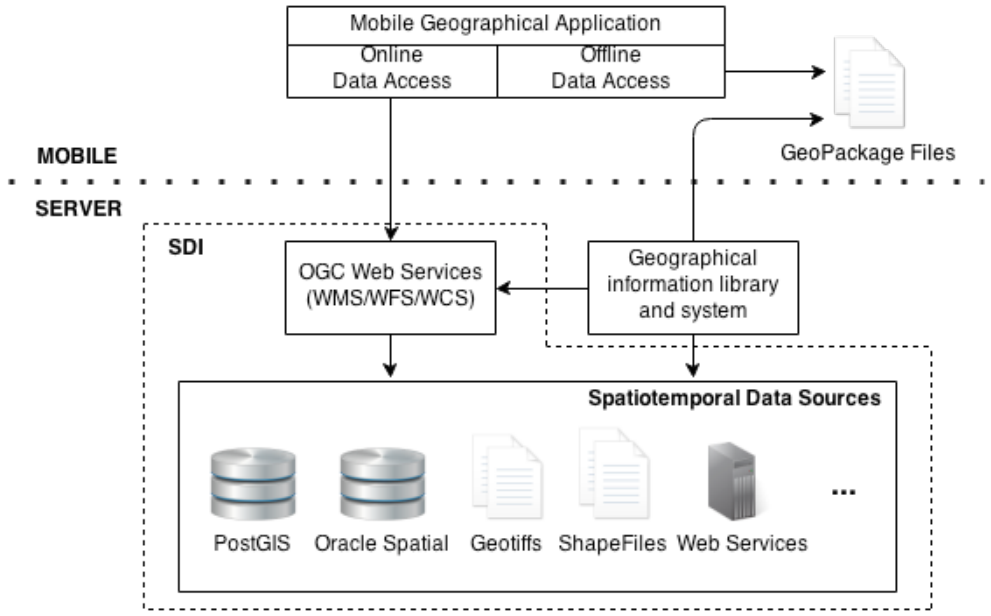


Figure 1. A Hybrid Architecture for Mobile Geographical Data Collection and Validation Systems

On the server side we need geographical information libraries and systems to decode and encode Geopackage files. These systems must be able to create Geopackage files from different kinds of spatiotemporal data sources, including files, spatial databases and web services. Users will use these systems to access different kinds of data sources, select the necessary spatiotemporal data sets from these sources to be used in the field, and export these data sets to Geopackage files. Users will execute this task of creating the necessary Geopackages files during the fieldwork planning phase.

4. Geopackage: Tests and Evaluation

As part of this work, we tested and evaluated the suitability of Geopackage files to support the interoperability between SDIs and the offline module of mobile systems for geographical data collection and validation. Mobile devices have limited storage capacity and processing power. Thus, we evaluated Geopackage documents by comparing their sizes and query processing times with other data sources.

We tested Geopackage files with *vector data* and *pyramid raster tiles* [OGC 2014]. Vector data is necessary for applications that need to handle feature geometries and attributes, for example, applications that execute spatial and attribute queries or edit the feature geometries. Pyramid raster tiles are useful for applications that display geographical information only as background layers. A pyramid structure organizes the raster tiles in a series of reduced/increased resolutions and is mainly used to improve the display performance.

4.1. Geographical Data Sets

For testing vector data, we created two data sets using the deforested regions detected by the projects DETER and PRODES, as shown in Figure 2. The first data set from DETER contains 439,596 regions or polygons detected from 2004 and 2012. The second one from PRODES contains 1,350,652 polygons detected from 2001 to 2012. From these two data sets, we generated three data sources; Geopackage vector files (GPKG), shapefiles (SHP) and PostGIS database (PG); and compared their sizes and query processing times. All these data sources were created with spatial indexes.

Since mobile devices have limited storage capacity, the files used in the "Offline Data Access" module must be as small as possible. Two possible options for this module are shapefile and Geopackage files; but we included PostGIS database running on local machine to illustrate our comparison. Figure 3 presents the sizes in megabytes (MB) of the three created data sources. The sizes of Geopackage files are 326 MB (DETER) and 1321 MB (PRODES), while shapefiles are 850 MB (DETER) and 2600 MB (PRODES). Geopackage vector files are around 50% smaller than shapefiles.

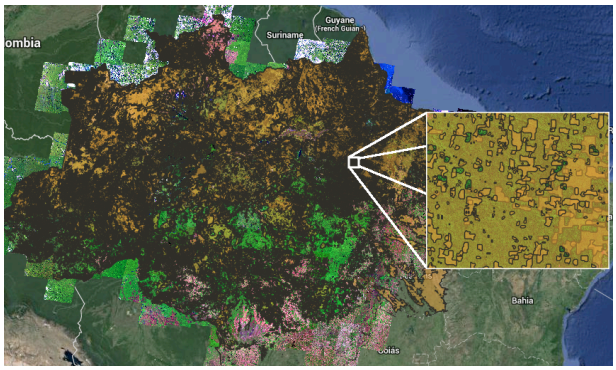


Figure 2. Data sets.

Figure 3. Data set sizes.

4.2. Attribute and Spatial Filter

Geographical data collection and validation applications can require operations that involve attribute and spatial filters. Examples of these operations are "*given a spatial location, return the attributes to be edited of the region that contains this location*" or "*return the regions whose areas are smaller than x*". Thus, we executed these two filters in the three created data sources and compared their runtimes. Runtime is the period for executing the query and fetching all returned records.

For attribute filter, we used the queries "*select all deforested regions detected by DETER project in 2003-01-01*" and "*select all deforested regions detected by PRODES project in 2005*". The first query returned 138,955 polygons and the second one, 241,439. The runtimes (in milliseconds) of these queries in the three data sources are presented in Table 1 and Figure 4.

For spatial filter, we used the bounding box of the Legal Amazon and selected all deforested regions inside this box. This query returned 430,044 polygons in the DETER data set and 1,300,552 polygons in the PRODES data set. The runtimes (in milliseconds) of these queries are presented in Table 2 and Figure 5.

Table 1. Attribute filter runtimes

Attribute Filter (Time ms)		
Data Source	**DETER**	**PRODES**
GPKG	1866	13700
PG	6245	12264
SHP	4163	29298

Table 2. Spatial filter runtimes

Spatial Filter (Time ms)		
Data Source	**DETER**	**PRODES**
GPKG	4197	23229
PG	11883	47902
SHP	3554	37423

Each runtime presented in Tables 1 and 2 is the average of ten iterations and all data sources were stored in the local machine. We can note that, in most cases, attribute and spatial filter runtimes in the Geopackage vector files are smaller than in the other data sources.

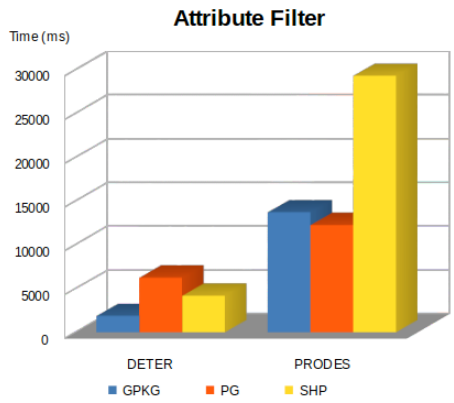


Figure 4. Attribute filter

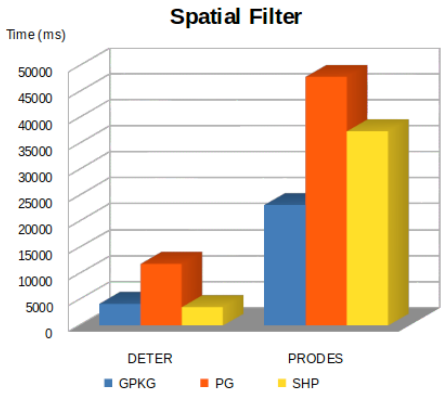


Figure 5. Spatial filter

4.3. Geopackage Tiles

For testing pyramid raster tiles, we created a Geopackage raster file from the deforested regions detected by PRODES project and Landsat 5 satellite images covering the Brazilian Legal Amazon. We created this file using tiles of 256 x 256 pixels and a pyramid with 10 levels of resolution. Figure 6 presents the Geopackage raster file, showing its level 7 in the big picture and its level 10 in the zoon area. Figure 7 shows the sizes in megabytes (MB) of the Geopackage vector and raster files. The vector file size is 1321 MB, while raster file is 552 MB.

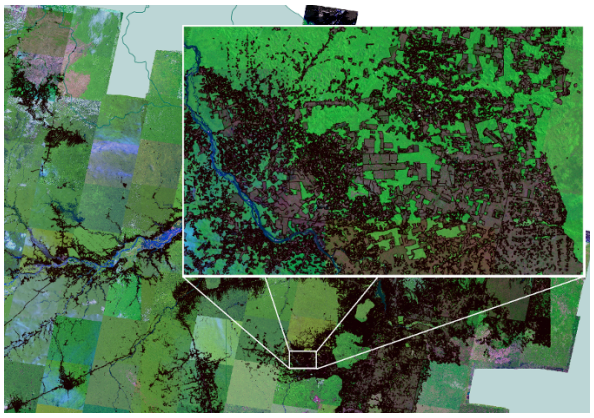


Figure 6. Geopackage tiles

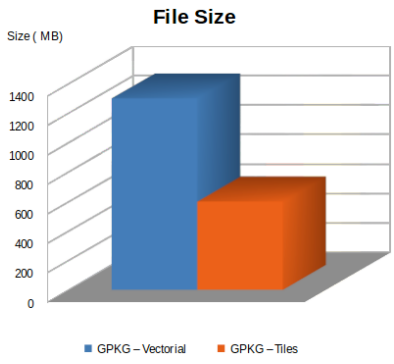


Figure 7. Geopackage file sizes

5. Final Remarks and Conclusions

This paper presents an ongoing work on designing and implementing a hybrid architecture for mobile geographical data acquisition and validation systems that can operate *online* as well as *offline*. We propose the use Geopackage documents as interoperable files between SDI and these mobile systems and show that this is viable through tests using Geopackage files with vector data and pyramid raster tiles.

In the tests, we used the GDAL/OGR library[1] to generate the Geopackage vector files and to access and execute the spatial and attribute filters in the three data sources. We also used the Mobile Atlas Creator[2] program to create the pyramid raster tiles from WMS server and the GeoTools library[3] to implement a JAVA code that stores these tiles as Geopackage raster files. The codes used in the tests are available in the GitHub: https://github.com/claudiobogossian/geopackage-test and https://github.com/claudiobogossian/GPKGTilesTest

As future work, we intend to develop a TerraView plugin that allows users to delimit an interesting area, access spatiotemporal data sets from different kinds of data sources and generate Geopackage files from these data sets. This plugin will be used in the server side and will play an important role in the proposed architecture, as described in Section 2. TerraView is a general-purpose GIS developed using the TerraLib GIS [Camara et al. 2008]. TerraView supports the development of plugin to enhance its functionalities.

References

Camara, G.; Vinhas, L.; Queiroz, G. R.; Ferreira, K. R.; Monteiro, A. M. V.; Carvalho, M. T. M.; Casanova, M. A. (2008) "TerraLib: An open-source GIS library for large-scale environmental and sócio-economic applications". *Open Source Approaches to Spatial Data Handling*. Berlin: Springer-Verlag.

Davis, C. A., Vellozo, H. S., Pinheiro, M. B. (2013) "Framework for Web and Mobile Volunteered Geographic Information Applications". *In: Proceedings of XIV Brazilian Symposium on Geoinformatics (GeoInfo 2013)*, November 24-27, 2013, Campos do Jordão, Brazil.

INPE (2014) Monitoramento da Floresta Amazônica Brasileira por Satélite (Monitoring the Brazilian Amazon Forest by Satellite). Available at www.obt.inpe.br/prodes.

Mohammadi, H. (2008) "The Integration of Multi-source Spatial Datasets in the Context of SDI Initiatives" *PhD thesis*, University of Melbourne. Available at: http://www.csdila.unimelb.edu.au/publication/theses/hossein-PhD.pdf (accessed in July 2014)

Open Geospatial Consortium – OGC. (2006) "OpenGIS Web Map Server Implementation Specification". Available at: http://www.opengeospatial.org/

Open Geospatial Consortium – OGC (2010) "OpenGIS Web Feature Service 2.0 Interface Standard". Available at: http://www.opengeospatial.org/

[1] Available at: http://www.gdal.org/

[2] Available at: http://mobac.sourceforge.net/

[3] Available at: http://www.geotools.org/

Open Geospatial Consortium – OGC (2014) "GeoPackage Encoding Standard". Available at: http://www.opengeospatial.org/

Poorazizi, E., Alesheikh, A. A., Behzadi, S. (2008) "Developing a Mobile GIS for Field Geospatial Data Acquisition". *Journal of Applied Sciences*, 8(18), 3279-3283.

Rajabifard, A., Feeny, M. E., Williamson, I. (2002). "Future Directions for SDI Development". *International Journal of Applied Earth Observation and Geoinformation* 4 (1), 11-22.

Raper, J., Gartner, G., Karimi, H., Rizos, C. (2008) "Applications of location–based services: a selected review". *Journal of Location Based Services*, 1(2), 89-111.

Tsou, M. H. (2004) "Integrated mobile GIS and wireless internet map servers for environmental monitoring and management", In: Special issue on Mobile Mapping and Geographic Information Systems, *Cartography and Geographic Information Science* 31 (3): 153–165.

Application of Geostatistical Conflation Techniques to Improve the Accuracy of Digital Elevation Models

Carlos A. Felgueiras, Jussara O. Ortiz[1], Eduardo C. G. Camargo[1]

[1]Divisão de Processamento de Imagens - Instituto Nacional de Pesquisas Espaciais (INPE) -
Caixa Postal 515 – 12201-970 – São José dos Campos – SP – Brazil

{carlos,jussara,eduardo}@dpi.inpe.br

Abstract. *This short paper describes and analyzes the results of a methodology that allows to conflate existing Digital Elevation Models with a sample set of elevation points in order to obtain more accurate results on modeling elevation information. The set of elevation points has higher vertical accuracy than the DEM and it is used the geostatistical procedure, known as kriging with an external drift, to perform the conflation. An initial case study is presented integrating a Shuttle Radar Topography Mission - SRTM - data and a sample set of elevation points obtained from a region of Campinas, city of São Paulo in Brazil. Others procedures of estimations and simulations will be considered in the future to explore the potential of the conflation techniques using geostatistics.*

1. Introduction

Digital Elevation Models - DEMs -, and their derivative products, are very important information used as input for spatial models performed in Geographical Information Systems - GIS - environment [Burrough 1987]. From a DEM it is possible to derive slope and aspect maps, drainage networks, contour lines, profile and volume calculations, etc. Nowadays it is possible to find DEM information for free, without financial costs, of almost any region of the earth surface. Unfortunately, the vertical (the heights) accuracy of these free DEMs are not appropriated for some spatial models. On the other hand elevation information of the earth surface can also be obtained in a set of spatial locations, 3D points, sampled in a geographical region of interest. These samples can be collected with very high vertical accuracy using Global Positioning System - GPS - equipments, for example.

Geostatistical tools has been used successfully to analyze and to model environmental attributes represented as a set of sample points of geophysical and geochemical indices, concentrations of soil elements, elevations, temperatures, etc. [Isaaks and Srivastava 1989, Goovaerts 1997]. Therefore, geostatistics can be applied to a sample set of elevation points, hereinafter referred to as "sample points", in order to create DEMs using estimations and simulations procedures. Also the geostatistical procedures allow performing conflations by integrating different sources of environmental attributes [Hengl et al 2008, Karkee et al 2008]. In the case of elevations there are kriging and simulation procedures that can be used to conflate existing DEMs

with sample points of the same geographical region. The objective of this conflation is to get a more accurate final DEM compared with the original, or the input, DEM.

In this context, the objective of this short paper is to describe and analyze a methodology to conflate DEMs with a sample set of elevation points in order to obtain more accurate results on modeling elevation information. The set of sample points has higher vertical accuracy than the original DEM and it is used the geostatistical procedure, known as *kriging with an external drift* - KED -, to perform the conflations. A case study is presented over a region of Campinas, city of São Paulo State in Brazil, to illustrate the application of the methodology to a real information of the earth surface.

2. Concepts and Methodology

2.1. Main Concepts

Geostatistical approaches for estimations and simulations are based on previous analysis of the spatial correlation of a set of sample points to represent the spatial variability, spatial dependence, of the attribute in a geographical region. This variability in function of the spatial distance is represented by variogram models. Empirical , or experimental, variogram models, $2\gamma^*(\mathbf{h})$, can be estimated, directly from a set of sample points, according to Equation 1 below:

$$2\gamma^*(\mathbf{h}) = \frac{1}{\mathrm{N}(\mathbf{h})} \sum_{(i,j)/\mathbf{h}_{ij} \approx \mathbf{h}} \left(z(\mathbf{u}_i) - z(\mathbf{u}_j)\right) * \left(z'(\mathbf{u}_i) - z'(\mathbf{u}_j)\right) \quad (1)$$

where $z(\mathbf{u}_i)$ and $z(\mathbf{u}_j)$ are attribute values observed at spatial locations $\mathbf{u}_i$ and $\mathbf{u}_j$ separated by the distance **h.** N(**h**) is the number of samples found inside a circumference with radius distance approximately equal to **h**.

The kriging procedure allows to infer a mean value of the attribute, in any spatial location **u**, from a number $n(\mathbf{u})$ of neighbor samples $z(\mathbf{u}_\alpha)$, α=1,...,$n(\mathbf{u})$. The general formulation for the kriging estimator is:

$$z^*(\mathbf{u}) - \mu(\mathbf{u}) = \sum_{\alpha=1}^{n(\mathbf{u})} \lambda_\alpha(\mathbf{u}) \cdot \left[z(\mathbf{u}_\alpha) - \mu(\mathbf{u}_\alpha)\right] \quad (2)$$

where $\mu(\mathbf{u}) = m(\mathbf{u})$ is the tendency, or the mean value, of the attribute in the spatial location **u**, $\mu(\mathbf{u}_\alpha)$ is the mean value in each sampled location $\mathbf{u}_\alpha$. The weights $\lambda_\alpha(\mathbf{u})$ are estimated considering the correlation structure defined by the modeled variogram for the set of sample points considered [Isaaks and Srivastava 1989].

Also, geostatistical approaches, for estimations and simulations, allow incorporating secondary information, along with primary one, in order to obtain more accurate and reliable models of the attribute. Variations of the kriging, using secondary information, can be: cokriging, universal kriging, regression kriging, kriging with varying local means, kriging with a trend model, kriging with an external drift, etc. [Ortiz et all 2007, Wackernagel 1998].

The KED approach is an extension of the kriging with a trend model, where the trend is obtained from a secondary (external) variable related to the primary one. The trends are used as the mean values, in the Equation 2, and the residual covariance, rather

than the covariance, of the primary variable must be used to solve this kriging approach. [Deutsch and Journel 1998] point out two conditions that have to be met before applying the external drift algorithm: (1) The external variable must vary smoothly in space and (2) The external variable must to be known at all locations $\mathbf{u}_\alpha$ of the primary data and at all locations **u** to be estimated. The advantage in this case is that it is not necessary to know the cross covariance between the primary and secondary variables. In this work the input DEM is considered the external drift yelling elevation data at all locations of the condition (2) above.

2.2. Methodology

The methodology of this work is based on the *kriging with an external drift* procedure presented in [Deutsch and Journel 1998] and follows the steps below:

1. Import a SRTM file from the region of study
2. Import the high vertical accurate sample set of elevation points
3. Create a residual file of the sample points taken the trend off the sample set
4. Create empirical and theoretical variograms for the set of sample points
5. Generate a DEM using the set of sample points and its theoretical variogram
6. Use the SRTM to create an ASCII data file with collocated elevation values and the residuals
7. Generate a conflated DEM using the SRTM data as gridded external drift and the collocated SRTM and the residuals information.

3. Results and Analysis

As a case study it was chosen a small region of Campinas, city of São Paulo State in Brazil. The geographical location coordinates of the bounding box of this region are: (w 47^o 08', s 23^o 00') e (o 46^o 57', s 22^o 50'"). The SRTM DEM information of the Campinas region is shown in the Figure 1.

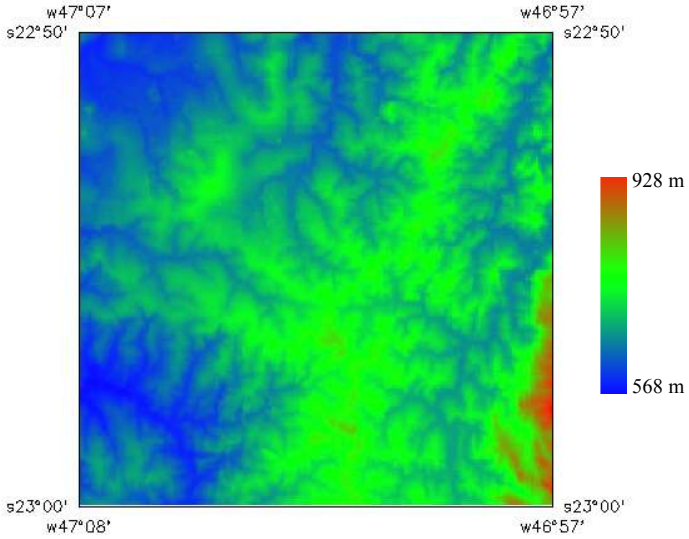


Figure 1. The SRTM DEM information of the Campinas region

The height resolution of this information, inside this region, is approximately 9.3m as assessed by analysis considering an accurate test set of 4549 elevation points.

In this experiment, it was considered a sample set of 505 elevation points sparsely distributed in the Campinas region. The height accuracy of the samples is 2.5m.

It was used geostatistical tools of the GIS known as SPRING, [Camara et al 1996] to perform most of the analysis and the estimations presented in this work. These tools were implemented in SPRING, [Camargo 1998], based on the original functions of the GSLib software developed by [Deutsch and Journel 1997]. Also, the GSLib was used to run the KED estimations to accomplish the conflation results.

Initially, the spatial locations of the sample points were used to obtain collocated elevation values from the SRTM information. This was performed by an external C program, developed for this purpose, that calculates also the residuals subtracting the elevation of the samples from the collocated SRTM elevation values.

Spatial dependence analysis of the sample points and of the residuals was performed to obtain empirical semivariograms. Theoretical semivariogram models (red curves) were fitted over these empirical semivariograms, as shown in Figure 2, representing the spatial variability of the elevations of the sample points (left) and of the residuals (right).

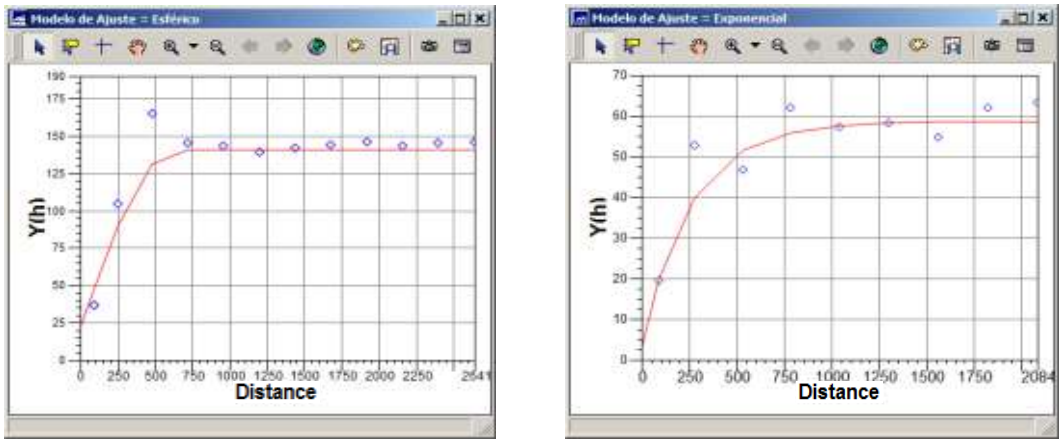


Figure 2. Theoretical semivariograms (red curve) fitted over empirical semivariograms (blue points) of the sample points (left) and of the residuals (right)

The theoretical variogram of the sample points is a spherical model with 21.48 m of nugget effect, 118.86 m of contribution and 645.46 m of range. The theoretical variogram of the residuals is an exponential model with 2.62 m of nugget effect, 55.82 m of contribution and 776.18 m of range.

The ordinary kriging procedure of the SPRING software was applied over the sample set of elevation points using the left theoretical variogram above. The resulting sample DEM, an elevation grid of size 200 columns by 200 rows, is shown in Figure 3. The spatial resolution of this grid is 90 meters in both x and y directions.

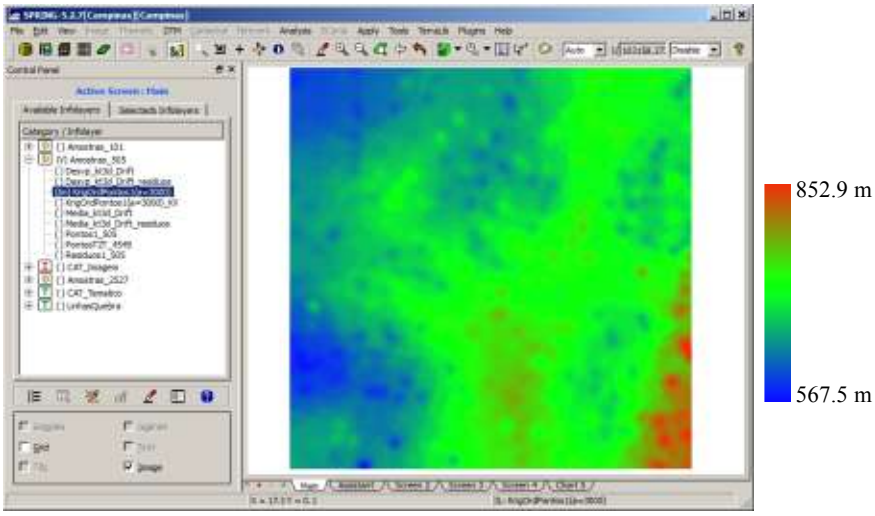


Figure 3. Digital Elevation Model estimated using only the sample points and their theoretical semivariogram

Following this, the kriging procedure of the GSLIB, known as K3td, was applied to the sample points, merged with collocated information of SRTM data. It was used theoretical semivariogram model of the residuals and the option of kriging with an external drift was chosen as parameters for this function. The gridded information of the SRTM was taken as the external drift file required for this KED procedure. The resulting DEM, an elevation grid of size 200 columns by 200 rows, is shown in Figure 4. The spatial resolution of this grid is 90x90 m in x and y directions.

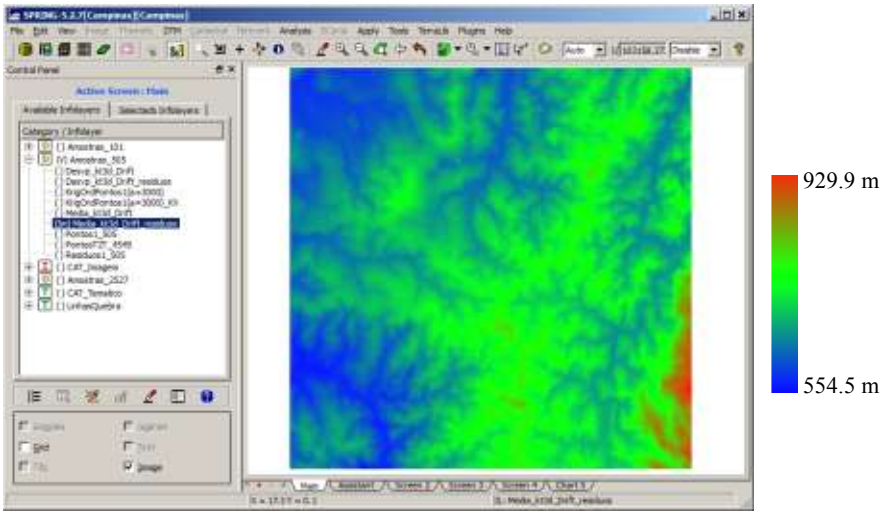


Figure 4. Digital Elevation Model obtained by conflation of the sample points and the SRTM data

A qualitative analysis, by visual inspection, between the maps of the Figures 1 and 3 shows that the SRTM DEM presents more detailed information. The map of Figure 3 is more homogeneous because the number of sample points is low, or it is not enough to represent the heterogeneity presented in the SRTM DEM .

A qualitative analysis, also by visual inspection, presents no differences between the DEMs of Figures 1 and 4. On the other hand, a quantitative analysis by a validation procedure, using a test set of 4549 elevation points, has shown that the vertical accuracy, measured by the standard deviation value, for the SRTM DEM is 9.32 meters.

When the same validation approach is applied to the DEM of the Figure 4 the calculated standard deviation value is 8.75 meters. This represents an increasing of 5.85% in accuracy of the final map (9.32-8.75)/9.32=0.0585. Table 1 presents other results obtained with different sets of sample points showing that the vertical accuracy increases with the number of sample points considered.

Table 1. Vertical DEM accuracy considering different number of sample points

Number of Samples (Tests) Points	Ordinary Kriging DEM (m)	SRTM DEM (m)	External Drift Kriging DEM (m)	Accuracy Improvement (%)
101 (4953)	34.53	9.28	9.09	2.20
505 (4549)	20.22	9.32	8.75	5.85
2527 (2527)	9.73	9.33	7.63	17.79

4. Conclusions

This article has shown that it is feasible to apply successfully the conflation technique using the geostatistical procedure known as kriging with an external drift. In this approach the SRTM DEM is considered the tendency of the information that is updated, or improved, with the information of a better accurate sample set of elevation points.

As shown by the case study it was obtained an improvement on the vertical resolution of 5.85% when it was used 505 sample points along with the SRTM considered. The improvement is better when the number of sample points is larger, as shown in Table 1. The results could be improved if a sample set with a better vertical resolution was used in this work.

This short paper presents only a small prototype representing the beginning of the researches related to the conflation techniques to improve existing DEMs.

Other spatial regions and geostatistical options will be considered in the future such as: cokriging and co-simulations, kriging and simulations with regression and with varying local means and the same variations for indicator krigings and simulations.

References

Burrough, P. A. (1987) "Principles of geographical information systems". New York: Oxford University Press. 344p.

Camara, G., Souza, R. C. M., Freitas U. M. and Garrido, J. (1996) "SPRING, Integrating Remote Sensing and GIS by object-oriented data modeling". Computer & Graphics, v. 20, n. 17, p.395-403.

Camargo, E. C. G. (1997) "Desenvolvimento, implementação e teste de procedimentos geoestatísticos (Krigeagem) no sistema de processamento de informações georreferenciadas (SPRING)". 1997-06. 146 p. (INPE-6410-TDI/620). Available at: <http://urlib.net/sid.inpe.br/iris@1912/2005/07.20.08.47.41>.

Deutsch, C. V. and Journel, A. G. (1998) "GSLIB: geostatistical software library and user's guide". New York: Oxford University Press, 369p.

Goovaerts, P. (1997) "Geostatistics for natural resources evaluation". New York: Oxford University Press, 483p.

Hengl, T., Bajat, B., Blagojevic, D., Reuter, H. I. (2008) "Geostatistical modeling of topography using auxiliary maps". ;Computers & Geosciences, 1886-1899.

Isaaks, E. H; Srivastava, R. M. (1989) "An introduction to applied geostatistics". New York: Oxford University Press, 561p.

Karkee, M.; Steward, B. L.; Azis, S. A. (2008) "Improving quality of public domain digital elevation models through data fusion". Biosystems Engineering, v.101, p.293-305, 2008.

Ortiz, J.O., Felgueiras, C.A.; Druck, S.; Monteiro, A.M.V. (2007) "Avaliação do procedimento de geoestatística de cokrigeagem para determinação da distribuição espacial de propriedades do solo". Geomática, Embrapa, cap. 7.

Wackernagel, H. (1998) "Multivariate Geostatistics". New York, Springer, 291p.

A RDF Vocabulary for Spatiotemporal Observation Data Sources

Karine Reis Ferreira [1], Diego Benincasa F. C. Almeida[1], Antônio Miguel Vieira Monteiro[1]

[1]DPI – Instituto Nacional de Pesquisas Espaciais (INPE)

{karine, benincasa, miguel}@dpi.inpe.br

Abstract. *Observations are our means to assess spatiotemporal phenomena in the real world. They are basic units for spatiotemporal data representation and are distributed by data providers using different formats and standards. In this work, we propose an approach to discover, access and integrate spatiotemporal observations from different kinds of data sources using RDF framework and SPARQL language. This paper presents an ongoing work on defining a RDF vocabulary for describing spatiotemporal observation data sources.*

1. Introduction

The recent technological advances in geospatial data collection, such as Earth observation and GPS satellites, have created massive data sets with better spatial and temporal resolution than ever. This scenario has motivated a challenge for Geoinformatics. We need geographical information systems (GIS) that can access spatiotemporal data sets from different kinds of data sources and analyze them in an integrated way.

Observations are our means to assess spatiotemporal phenomena in the real world. Although most spatiotemporal phenomena are continuous over time and space, they are often measured through discrete observations. Observations link information to reality and provide the building blocks of conceptualizations [Kuhn 2009]. Recent research draws attention to the importance of using observations as a basis for designing geospatial applications [Kuhn 2005]. Following this trend, we proposed a data model for spatiotemporal data representation grounded on observations [Ferreira et al. 2014] and implemented it in the TerraLib GIS library [Camara et al. 2008]. Taking observations as basic units for spatiotemporal data representation, this work focuses on *how to access and combine observations from different kinds of data sources*.

RDF (Resource Description Framework) is a data model for describing and connecting resources and SPARQL is a query language for RDF data sets. Both are World Wide Web Consortium (W3C) standards and are key techniques in *Linked Data* and *Semantic Web* [Berners-Lee et al. 2001] [Wood et al. 2014]. The term Linked Data refers to a set of best practices for publishing and connecting structured data on the Web using international standards of W3C. Semantic Web provides technologies that allow data to be shared with explicit meaning and processed by machines. Publishing information as linked data is the first step towards the world of semantic web [Herman 2012]. In this work, we propose an approach that uses RDF to describe data sources that provide spatiotemporal observations and SPARQL to discover information about these data sources.

RDF describes resources using the concepts of classes, properties, and values. The term *vocabulary* refers to a set of classes and properties that are defined specifically for a certain application. RDF Schema (RDFS) is a framework to define such vocabularies, that is, to describe application-specific classes and properties. Examples of vocabularies are Dublin Core Vocabulary[1] that defines a set of predefined properties for describing documents and FOAF (Friend Of A Friend) Vocabulary[2] that describe relationships among people. This paper presents an ongoing work on defining a RDF vocabulary for describing spatiotemporal observation data sources.

2. Related Work

In a previous work, we propose an approach to access trajectories of moving objects from different kinds of data sources based on XML metadata files [Ferreira et al. 2013]. Differently, this paper presents a new approach to access spatiotemporal observations from different kinds of data sources, using RDF framework and SPARQL language. The new approach is more comprehensive than the previous one because it is for spatiotemporal observation data sources and not only for trajectory data sources. Besides that, the new approach uses RDF files to describe data sources. It allows the use of SPARQL language on these files to discover information about the data sources.

There are many RDF vocabularies for different application domains, such as Dublin Core for describing documents and FOAF for describing relationships among people. Examples of RDF vocabularies for describing geospatial data are W3C Basic Geo vocabulary, GeoOWL ontology, NeoGeo Vocabulary and GeoSPARQL [Battle and Kolas 2012]. GeoSPARQL is an Open Geospatial Consortium (OGC) standard that defines a vocabulary for representing geospatial data in RDF and an extension to the SPARQL query language for processing geospatial data [OGC 2012a].

RDF framework is the key technique in Linked Data and Semantic Web. Many initiatives and projects focus on transforming geospatial data into linked RDF files, such as the LinkedGeoData and Geonames.org[3] [Stadler et al. 2012] [Janowicz et al. 2012]. The LinkedGeoData project provides a RDF serialization of Points Of Interest from Open Street Map. Geonames.org provides a database that contains over 10 million geographical names and an interface to generate a RDF-dump of this database. In our proposal, we use RDF as linked metadata files, that is, files that describe *how* data sources represent spatiotemporal observations and *links* among these data sources. In this first step, we will not transform the spatiotemporal observations from their original data sources and formats into RDF files.

3. An Observation-Based Model for Spatiotemporal Data

We proposed a data model for spatiotemporal information grounded on observations and specified it using an algebraic formalism [Ferreira et al. 2014]. Algebras describe data types and their operations in a formal way, independently of programming languages. By separating specification from implementation, they help to develop interoperable, reliable and expressive applications [Frank 1999]. The proposed algebra

[1] http://dublincore.org/
[2] http://xmlns.com/foaf/spec/
[3] http://www.geonames.org/

is extensible, defining data types as *building blocks* for other types, as shown in Figure 1.

The proposed model defines three spatiotemporal data types are defined as abstractions built on observations: *time series*, *trajectory*, and *coverage* [Ferreira et al. 2014]. A *time series* represents the variation of a property over time in a fixed location. A *trajectory* represents how locations or boundaries of an object change over time. A *coverage* represents the variation of a property in a spatial extent at a time. We also define an auxiliary type called *coverage series* that represents a time-ordered set of coverages that have the same boundary. Using these types, we can construct *objects* and *events*.

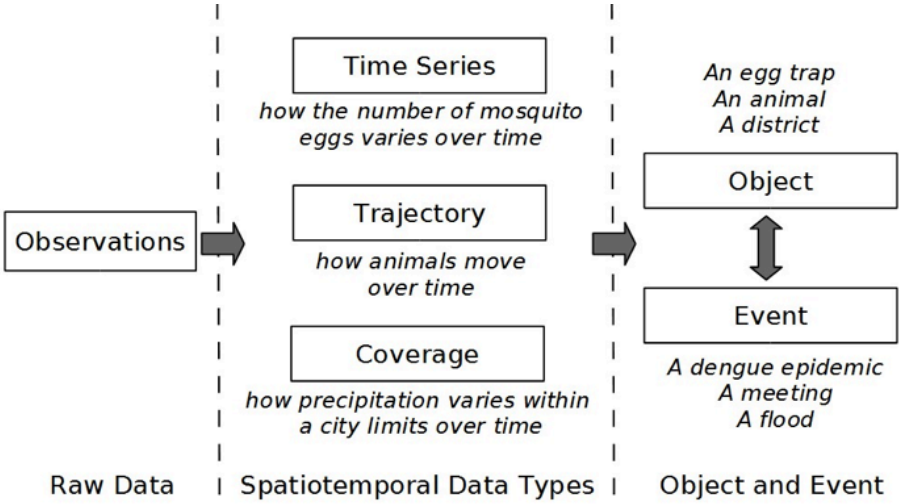


Figure 1. An Algebra for Spatiotemporal Data: From Observations To Events. Source [Ferreira et al. 2014]

Using these three spatiotemporal data types, we can create different views on the same observation set, meeting the application needs. Consider a set of cars equipped with GPS and air pollution sensors. Figure 2 shows tracks of three cars in a city during one day. These cars produce an observation set, where each one contains a car identity, a time instant, a location and an air pollution value. The observations are taken at each hour. From this observation set, we can extract information using different kinds of queries, such as *how the average air pollution varies over time in the city*; *how the cars move over time and space*; and *how pollution varies within the city limits*.

Each application needs different queries and each kind of query is suited to a specific data type. Taking the whole city as a fixed reference, we can get a *time series* that represents the variation of the average air pollution in the city per hour. Considering each car an individual object, we can get a set of *trajectories*. Making the whole day as a time reference and taking all observations at that day, we can create a *coverage* to represent the air pollution variation within the city limits during that day.

Algebraic specifications are language-independent. Programmers can translate them into software using programming languages of their choice. We implemented the proposed algebra in the TerraLib GIS library [Camara et al. 2008]. We developed a new module called “TerraLib ST module” to deal with spatiotemporal information that contains all data types and operations described in the proposed algebra.

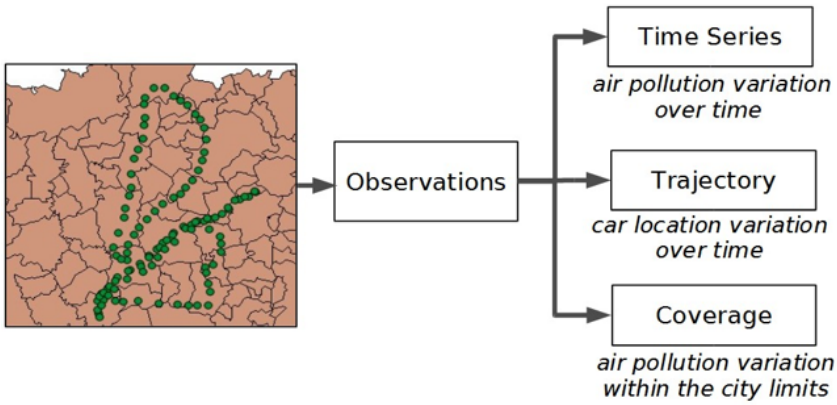


Figure 2. Different views on observations produced by moving cars. Source [Ferreira et al. 2014]

4. The Proposal

Spatiotemporal observations can be stored and disseminated by data providers in different ways. Recently, most data providers throughout the world have organized their geospatial data sets and made them available on the Internet via files, spatial databases and web services, following well-established standards[4] defined by ISO and OGC. Geography Markup Language (GML) and Keyhole Markup Language (KML) are examples of OGC's file formats for spatial data interchange. Spatial extensions of traditional database management systems, such as PostGIS and Oracle Spatial, deal with vector spatial information in compliance with the OGC Simple Feature Access (SFA) specification. Besides that, there are standards for serving spatial data and processes via web services such as Web Feature Service (WFS) and WCS (Web Coverage Service).

OGC proposes a standard called Sensor Observation Service (SOS) that defines a web service interface for disseminating and querying observations, sensor metadata and observed features [OGC 2012b], based on the OGC Observations and Measurements (O&M) specification [OGC 2010]. However, many data providers store and disseminate spatiotemporal information using other formats and standards, not only SOS, as shown in Figure 3. GIS tools must be able to access different types of spatiotemporal data sources, without forcing the use of a specific format or standard.

Therefore, we propose an approach to access spatiotemporal observations from different kinds of data sources using RDF framework and SPARQL language. The central idea is to use RDF files for describing how spatiotemporal observations are represented in data sources and SPARQL language for discovering information about these data sources. Each data source has an associated RDF file and all RDF files are based in the same vocabulary, as presented in Figure 3. We are working on defining and implementing a RDF vocabulary for spatiotemporal observation data sources. The main features of this vocabulary are:

1. It will be based on the observation concept defined in the OGC O&M specification [OGC 2010]. So, it must contain classes and properties to describe

[4] http://www.opengeospatial.org/standards/is

how phenomenon time, result time, valid time, and observed attributes are represented in different kinds of data sources.

2. It will use the OGC GeoSPARQL schema [OGC 2012a] to represent spatial information in RDF files. This is important, for example, to encode observation spatial extents in RDF files.
3. It must support link among data sources. Two RDF files that describe data sources can be linked. For example, the observations can be stored in a data source and their spatial extent can be stored in another data source. In this case, the RDF file that describes the first data source must have a link to the RDF file that describes the second one.

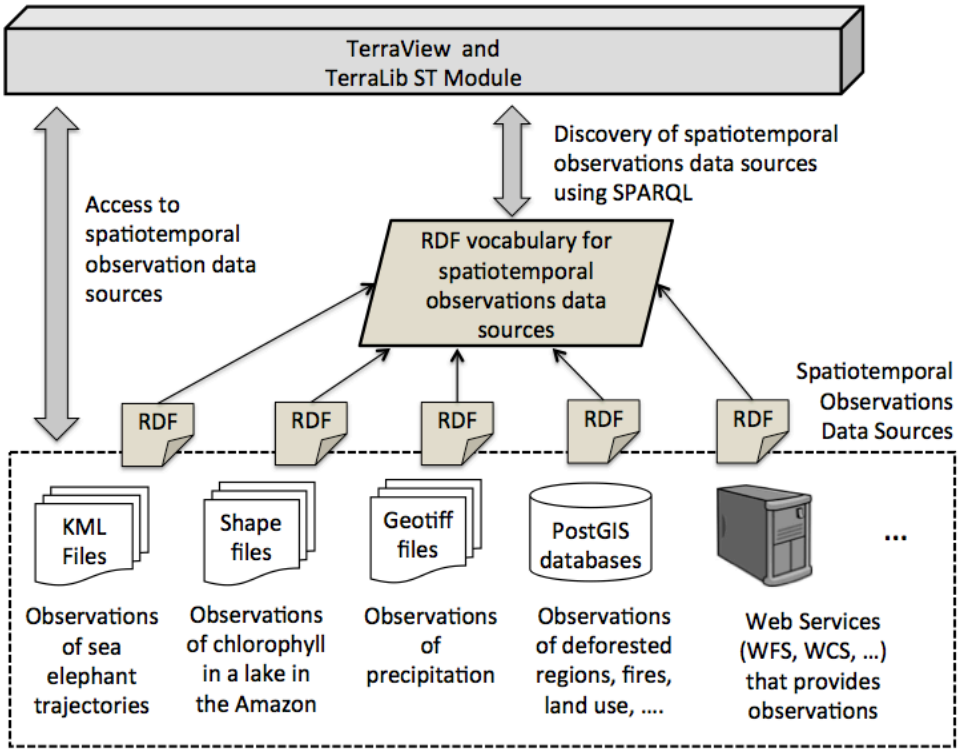


Figure 3. The proposal

The TerraLib ST Module contains the data types and operations defined in the model presented in Section 3. It is responsible for discovering information about the spatiotemporal observation data sources using SPARQL on the RDF files and accessing these observations. After loading observations from data sources, users can create the most suitable views on them, as shown in Figure 2, in the application level. Observations are basic units without strong semantics. TerraView/TerraLib users decide what abstraction, *time series*, *trajectory* or *coverage*, will be created on these observations to fit the application needs.

5. Final Remarks

This paper presents an approach to discover, access and integrate spatiotemporal observations from different kinds of data sources using RDF framework and SPARQL language. This is an ongoing work. In this approach, we are working on defining a RDF

vocabulary for describing spatiotemporal observation data sources based on the OGC O&M and GeoSPARQL specifications.

Our proposal considers that data sources provide observations which are basic units for spatiotemporal phenomenon representation. This allows application users to create different views on observations, according to the application needs. This work is crucial in extending TerraLib GIS library and TerraView to deal with spatiotemporal data.

References

Battle, R. and Kolas D. (2012) "Enabling the geospatial semantic web with parliament and GeoSPARQL". *Semantic Web Journal* 3(4).

Berners-Lee, T.; Hendler, J.; and Lassila, O. (2001). The semantic web. *Scientific american*, 284(5), 28-37.

Camara, G.; Vinhas, L.; Queiroz, G. R.; Ferreira, K. R.; Monteiro, A. M. V.; Carvalho, M. T. M.; Casanova, M. A. (2008) "TerraLib: An open-source GIS library for large-scale environmental and sócio-economic applications". *Open Source Approaches to Spatial Data Handling*. Berlin: Springer-Verlag.

Ferreira, K. R.; Camara, G.; Monteiro, A. M. V. (2014) "An algebra for spatiotemporal data: From observations to events". *Transactions in GIS*, 18(2), 253-269.

Ferreira, K. R.; Vinhas, L.; Monteiro, A. M. V.; Camara, G. (2013) "Moving Objects and Spatial Data Sources". *Revista Brasileira de Cartografia*, (64/4).

Frank, A. U. (1999) "One step up the abstraction ladder: Combining algebras – from functional pieces to a whole". In Freksa C and Mark D (eds) *COSIT: Conference on Spatial Information Theory*. Berlin, Springer Lecture Notes in Computer Science, 1661, 95-108

Herman, I. (2012) "Tutorial on Semantic Web". Available at: http://www.w3.org/People/Ivan/CorePresentations/SWTutorial/

Janowicz, K.; Scheider, S.; Pehle, T.; Hart, G. (2012) "Geospatial semantics and linked spatiotemporal data–Past, present, and future". *Semantic Web Journal*, 3(4), 321-332.

Kuhn, W. (2005) "Geospatial Semantics: Why, of What, and How?". *Journal of Data Semantics*, 3(1).

Kuhn, W. (2009) "A functional ontology of observation and measurement". In Janowicz K, Raubal M, and Levashkin S (eds) *International Conference on GeoSpatial Semantics (GeoS 2009)*. Berlin, Springer Lecture Notes in Computer Science, 5892, 26-43

Open Geospatial Consortium – OGC (2012a) "OGC GeoSPARQL - A Geographic Query Language for RDF Data". Available at: http://www.opengeospatial.org/

Open Geospatial Consortium – OGC (2012b) "OGC Sensor Observation Service Interface Standard". Available at: http://www.opengeospatial.org/

Open Geospatial Consortium – OGC. (2010) "Geographic Information: Observations and Measurements". Available at: http://www.opengeospatial.org/

Stadler, C.; Lehmann, J.; Höffner K.; Auer S. (2012) "Linkedgeodata: A core for a web of spatial open data". *Semantic Web Journal* 3(4).

Wood, D.; Zaidman, M.; Ruth, L. (2014) *Linked Data – Structured Data on the Web*. Manning Publications.

Agentes de Mineração de Imagens de Satélite

Ciro D. G. Moura[1], Nicksson C. A. Freitas[2], Marcelino P. S. Silva[1]

[1]Programa de Pós-Graduação em Ciência da Computação, Universidade do Estado do Rio Grande do Norte (UERN) / Universidade Federal Rural do Semi-Árido (UFERSA) – Mossoró, RN – Brasil

[2]LES – Laboratório de Engenharia de Software, Departamento de Informática, Universidade do Estado do Rio Grande do Norte (UERN) – Mossoró, RN – Brasil

{ciro.dgm, nickssonarrais, prof.marcelino}@gmail.com

Abstract. *Image Mining is a field with huge potential and relevant challenges. For satellite images, this technique and resources as well algorithms can bring appropriate responses to important problems. However, due to this approach limitations, multiagent systems present features that, properly applied, can bring advances on pattern analysis in satellite images. In this context, the goal of this research is to present a methodology and a software for satellite image mining through multiagent systems. The prototype is being developed in Java and uses resources from TerraView and GeoDMA.*

Resumo. *Mineração de Imagens é uma área com grande potencial e relevantes desafios. No caso de imagens de satélite, esta técnica e seu conjunto de recursos e algoritmos podem viabilizar soluções e respostas adequadas a importantes problemas. Entretanto, devido às limitações desta abordagem, sistemas multiagentes possuem características que, adequadamente aplicadas, podem trazer avanços significativos na análise de padrões em imagens de satélite. Neste contexto, o objetivo deste trabalho é apresentar uma metodologia e respectiva implementação computacional para a mineração de imagens de satélite através de sistemas multiagentes. O protótipo está sendo desenvolvido em Java e utiliza funcionalidades do TerraView e o do GeoDMA.*

1. Introdução

A mineração de dados é uma das etapas no processo de Descoberta de Conhecimento em Banco de Dados (DCBD). Essa etapa é responsável pela aplicação de algoritmos específicos para extrair padrões de dados [Fayyad, 1997].

Na mineração de imagens, as imagens de um acervo (banco de imagens) são recuperadas segundo critérios inerentes à aplicação. A seguir, uma fase de pré-processamento aumenta a qualidade dos dados, os quais são então submetidos a uma série de transformações e de extração de características que geram importantes informações a respeito das imagens. A partir destas informações, a mineração pode ser realizada através de técnicas específicas, com o intuído de descobrir padrões significativos. Esses padrões resultantes são então avaliados e interpretados para a

obtenção do conhecimento final, que pode ser aplicado no entendimento de problemas, na tomada de decisões ou outras aplicações estratégicas [Silva, 2006].

Sistemas de agentes ou multiagentes e o processo de DCBD são duas áreas de pesquisa cada vez mais inter-relacionadas, que proporcionam benefícios para ambas as partes. Para Silva e Ralha (2011), essas duas áreas, a princípio, seguem com metas e objetivos distintos, porém há vários aspectos de ambas as áreas que coincidem, tais como: interação usuário-sistema, papéis humanos, modelagem dinâmica, fatores de domínio, fatores organizacionais e sociais, entre outros.

Nesse contexto, o objetivo deste trabalho é apresentar uma metodologia e respectiva ferramenta para a mineração de imagens através de sistemas multiagentes. O protótipo foi desenvolvido em Java e trabalha em conjunto com o software TerraView e o seu plugin, o GeoDMA.

2. Trabalhos Relacionados

A interação entre mineração de dados e agentes é aplicada em diferentes campos de pesquisa. Apresentaremos em seguida alguns trabalhos relacionados ao tema.

Silva (2011) apresenta um protótipo do AGent Mining Integration (AGMI) que foi testado com dados reais de licitações extraídos do Sistema ComprasNet. Vários experimentos foram realizados explorando aspectos de distribuição do processamento e autonomia dos agentes.

Pultar, Raubal, e Goodchild, (2008) trazem o projeto e a implementação de um protótipo que lê os dados de páginas na web e segue links para adquirir conhecimento. O agente cria um banco de dados com textos de páginas da web, minera informações de localização e então converte para um formato de dados geoespaciais.

Xavier Júnior (2012) propôs um Sistema Multiagente que permite que os usuários possam guardar as informações visualizadas em seus históricos de acesso, e baseado nesse histórico, o sistema cria agrupamentos de informações relevantes através do uso de métodos de agrupamento de dados relacionais. Com as informações armazenadas no perfil dos usuários, os agentes podem interagir e recomendar informações geográficas relevantes aos usuários.

Todos esses trabalhos exemplificam o uso de agentes de mineração de dados, mas nenhuma das metodologias é aplicada a mineração de imagens de satélite. Tendo como base conceitos abordados nesses trabalhos, este trabalho apresenta uma metodologia diferente.

3. Técnicas e Ferramentas

A metodologia proposta e descrita na próxima seção utiliza técnicas e ferramentas de mineração de dados geográficos e de sistemas multiagentes.

3.1 GeoDMA

A ferramenta utilizada para minerar imagens é o GeoDMA (Geographical Data Mining Analyst). Ela realiza todas as fases de processamento necessárias para manipular dados de sensoriamento remoto, incluindo os processos de segmentação, extração e seleção de

atributos, treinamento, classificação e análise exploratória dos dados. Segundo Korting et al. (2009), o GeoDMA funciona como um plugin para o sistema TerraView. Isto significa que toda a estrutura que manipula e visualiza bancos de dados geográficos é proporcionada pelo TerraView. Então o GeoDMA é executado em conjunto com o TerraView, e produz os resultados que são exibidos na sua tela principal.

Dessa forma, uma imagem é inserida no Terraview e, utilizando o GeoDMA, é possível segmentar a imagem e extrair atributos espaciais e espectrais de cada um desses segmentos. Após a etapa de extração, na etapa de treinamento seleciona-se alguns dos segmentos e seus respectivos atributos, rotulando cada um desses segmentos com uma determinada classe. Com as amostras selecionadas nesta etapa, o próximo passo é a classificação, cujo algoritmo escolhido para este trabalho foi a árvore de decisão C4.5 [Quinlan, 1993], uma vez que o GeoDMA pode carregar uma árvore já existente ou salvar uma nova para que ela possa ser usada em futuras classificações.

3.2 Sistemas Multiagentes

Um agente é capaz de perceber seu ambiente por meio de sensores e de agir sobre este ambiente por intermédio de atuadores. Os quatro tipos básicos de agentes são o agente reativo simples, o agente reativo com estado, o agente baseado em objetivo e o agente baseado em utilidade [Russel e Norving, 2010]. Um Sistema Multiagente (SMA) consiste de um número de agentes que interagem uns com os outros de forma cooperativa ou competitiva. No caso mais geral, os agentes em um SMA estarão representando ou agindo em nome de usuários ou proprietários com diferentes objetivos e motivações [Wooldridge, 2009].

Segundo Sycara (1998), as características de SMA são tais que: cada agente possui uma visão limitada; não há controle global do sistema; trabalham com dados descentralizados; a computação é assíncrona, de maneira a permitir a comunicação entre entidades heterogêneas. Para Weiss (2000) e Wooldridge (2009) uma das razões para o desenvolvimento da área é que os conceitos subjacentes à SMA não são restritos a um único domínio de aplicação, sendo interessantes no desenvolvimento e análise de modelos e teorias de interatividade nas sociedades humanas. Isto ocorre devido ao SMA ser considerado uma metáfora natural para o entendimento e construção de uma ampla faixa do que podem ser rudemente denominados de sistemas artificiais sociais.

Aplicando os conceitos de agentes e SMA neste trabalho, cada agente realiza uma busca por padrões nas imagens de acordo com seu perfil e, dessa forma, o SMA minera a imagem como um todo.

4. Metodologia

O GeoDMA realiza todo o processo para classificar uma única imagem. Visando superar esta limitação, esta proposta com sistemas multiagente viabiliza um processo articulado e cooperativo. Uma vez que o GeoDMA segmenta a imagem e extrai os atributos de cada um dos segmentos, armazenando-os no banco de dados, o protótipo realizará a mineração através dos agentes segundo a árvore de decisão gerada.

A Figura 1 apresenta o diagrama do processo. (1) O TerraView gerencia o banco de dados e manipula as imagens; (2) O GeoDMA segmenta as imagens e extrai os atributos espaciais e espectrais de cada um dos segmentos; (3) Esses atributos são

armazenados no banco de dados; (4) A ferramenta então é conectada ao mesmo banco de dados utilizado pelo TerraView, e as tabelas que possuem os atributos obtidos são selecionados para criar o ambiente dos agentes; neste ponto, o usuário cria ou escolhe os agentes que serão utilizados visando atingir os seus resultados esperados. Cada agente possui sua árvore de decisão para que os segmentos possam ser minerados pelo mesmo. (5) Finalmente, com os dados classificados, o usuário visualiza o resultado da mineração no TerraView.

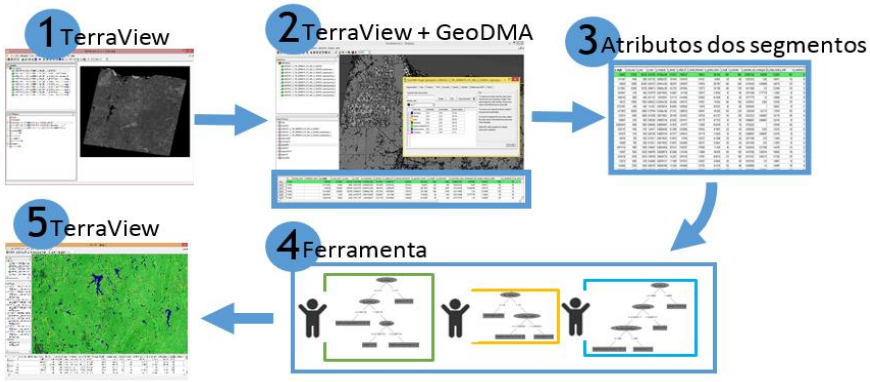


Figura 1. Diagrama do processo

Os agentes são do tipo reativos simples, ou seja, são agentes que selecionam ações com base na sua percepção atual [Russel e Norving, 2010]. A partir da sua árvore, cada agente possuirá um perfil que lhe permitirá minerar padrões específicos em novas imagens. Então, digamos que o agente A1 tenha o perfil de minerar corpos d'água. Na interface do protótipo o usuário terá recursos para selecioná-lo e atribuir ao mesmo a tarefa de minerar em novas imagens objetos que sejam mananciais, lagoas, açudes ou outros objetos pertinentes ao seu perfil.

Dessa forma, o agente A1 é capaz de identificar todos os segmentos que satisfaçam sua árvore de decisão, superando a limitação do GeoDMA, pois os agentes não minerarão apenas os segmentos de uma imagem, mas todos os segmentos das imagens que forem alocadas em seu ambiente.

5. Estudo de caso

Pereira (2010) realizou um estudo que tinha como objetivo mapear e identificar as áreas sujeitas ao processo de desertificação no estado do Rio Grande do Norte. Para isso, usou-se uma série histórica de imagens do satélite Landsat 5 TM dos anos de 1987, 1999 e 2008, e a cena 215/64. Foi realizado um recorte abrangendo 13 municípios: Jucurutu, Timbaúba dos Batistas, São Fernando, Florânea, Cruzeta, São José do Seridó, Acari, São Vicente, Tenente Laurentino Cruz, Lagoa Nova, currais novos, Bodó e Cerro Corá. Esses municípios foram escolhidos pois pertencem a uma região de grande susceptibilidade ao fenômeno da desertificação.

O software utilizado foi SPRING. Para segmentar as imagens, foi utilizada a técnica de crescimento de regiões, a qual agrega pixels com propriedades similares. Os valores de similaridade e área utilizados foram 10 e 15, respectivamente. Na etapa de classificação foi utilizado o classificador do tipo supervisionado "Bathacharia" e as seis

classes apresentadas nas imagens foram: caatinga preservada, caatinga degradada, solo exposto, lavoura, mata ciliar e corpos d'água.

Como forma de demonstrar a metodologia e o protótipo desenvolvido, realizamos uma comparação com o estudo apresentado por Pereira (2010), mas buscando padrões que indiquem a existência de corpos d'água. Para isso, utilizou-se a mesma série de imagens, obtidas no Banco de Imagens da DGI/INPE (Divisão de Geração de Imagens do Instituto Nacional de Pesquisas Espaciais. DGI, 2011). Apenas a banda 5 foi selecionada, pois corpos d'água são diferenciados de áreas com vegetação, áreas de cultivo, bem como solo exposto.

Utilizando o GeoDMA, as imagens foram segmentadas com a mesma técnica de crescimento de regiões e os mesmos valores para similaridade e área, a diferença fica na parte do treinamento, que é realizado apenas na imagem de 1987. Após a etapa de treinamento, uma árvore de decisão é gerada pelo GeoDMA para classificar a imagem de 1987. Essa árvore gerada serviu de perfil para um agente que foi o responsável pela classificação das outras duas imagens. A Figura 2 apresenta a comparação visual entre as imagens do trabalho de Pereira (2010) (A) e os resultados obtidos neste trabalho (B).

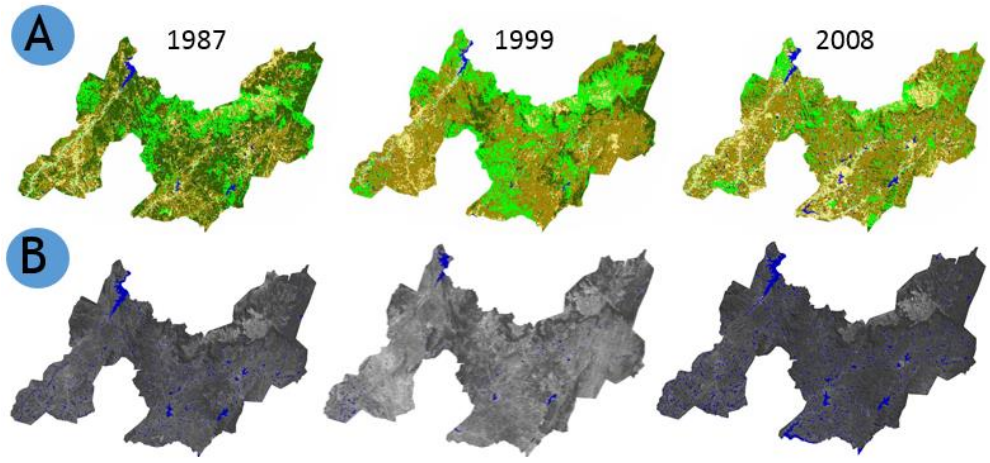


Figura 2. Resultado da classificação na área de corte: (A) Imagens adaptadas do trabalho da Pereira (2010) e (B) e imagens mineradas pelo agente.

Mesmo tendo utilizado softwares diferentes e diferentes amostras na etapa de treinamento, os resultados foram satisfatórios. Observa-se na Tabela 1 que a porcentagem de área ocupada pela classe água é bem próxima.

Tabela 1. Porcentagem da área ocupada pela classe água.

	1987	1999	2008
Pereira (2010)	1.94%	1.41%	3.12%
Resultados obtidos	2.09%	1.99%	3.36%

Da mesma forma como foi possível classificar utilizando um agente, outros agentes poderiam ter sido criados com diferentes perfis (lavoura, solo exposto etc.) e usados para minerar as imagens.

6. Conclusão

Este trabalho propõe uma metodologia e ferramenta que utiliza as características de agentes para auxiliar no processo de extração de conhecimento de imagens de satélite. Esses agentes serão capazes de minerar dados geográficos a partir das características espaciais e espectrais dos objetos presentes em imagens de sensoriamento remoto. O SMA, através dos diferentes perfis dos seus agentes, terá a funcionalidade, após devidamente treinados, de minerar objetos de diferentes padrões (solo exposto, vegetação densa, corpos d'água, dentre outros) em um repositório de imagens.

A ferramenta ainda está em desenvolvimento e, os resultados parciais estão sendo obtidos e avaliados através de protótipo. Novos estudos de caso validarão a versão funcional da ferramenta e, consequentemente, a metodologia proposta.

Referências

DGI. DGI - Divisão de Geração de Imagens (INPE), 2011. Catálogo de Imagens. Disponível em: <http://www.dgi.inpe.br/CDSR/>.

Fayyad, U. Data Mining and Knowledge Discovery in Databases: Implications for Scientific Datadabase. Proc. of the 9th International Conference on Scientific and Statistical Database Management, pp. 2-11, 1997.

Korting, T. S.; et al. GeoDMA - Um sistema para mineração de dados de sensoriamento remoto. In: SIMPÓSIO BRASILEIRO DE SENSORIAMENTO REMOTO, 14, 2009, Natal. Anais do XIV Simpósio Brasileiro de Sensoriamento Remoto. Natal: INPE, 2009. p. 7813-7820.

Pereira, G. O. Identificação de Áreas Suscetíveis à Desertificação no Rio Grande do Norte. Relatório final. (PIBIC/CNPq/INPE). INPE, 2010.

Pultar, E., Raubal, M. e Goodchild, M. F.. 2008. GEDMWA: Geospatial Exploratory Data Mining Web Agent. In Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS '08). ACM, New York, NY, USA, Article 74, 4 pages.

Quinlan, J. R. C4.5: Programs for Machine Learning. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993. ISBN 1-55860-238-0.

Russel, S. Norving, P. Artificial Intelligence: A Modern Approach. 3ª. Ed. Prentice Hall. 2010. ISBN-10: 0-13-604259-7.

Silva, C. V. S. Agentes de Mineração e sua Aplicação no Domínio de Auditoria Governamental. 2011. 122f.. Tese (Mestrado em Informática) - UnB, Brasília, 2011.

Silva, C. V. S. e Ralha, C. G. Detecção de Cartéis em Licitações Públicas com Agentes de Mineração de Dados. RESI - Revista Eletrônica de Sistema de Infomação. v. 10, n. 1, artigo 8. 2011. ISSN: 1677-3071.

Silva, M. P. S. Mineração de dados em bancos de imagens. 2006. 123f.. Tese (Doutorado em Computação Aplicada) - INPE, São José dos Campos, 2006.

Sycara, K. P.. Multiagent systems. AI Magazine, 19(2):79–92, 1998.

Weiss, G. Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. The MIT Press, 2000.

Wooldridge, M. An Introduction to MultiAgent Systems. John Wiley & Sons, LTD, Chichester, England, 2nd edition, 2009.

Xavier Júnior, J. C. NatalGIS: Um Sistema Multiagente de Recomendação de Informações Geográficas baseado em Agrupamento de Dados Relacionais. 2012. 189f.. Tese (Doutorado em Ciências) - UFRN, Natal, 2012.

Proposta de Sistema de monitoramento da sigatoka-negra baseado em variáveis ambientais utilizando o TerraMA^2

Hugo N. Bendini[1], Wilson S. Moraes[2], Simone S. da Costa[3], Eymar S. S. Lopes[4], Thales S. Körting[4], Leila M. G. Fonseca[4]

[1]Instituto Nacional de Pesquisas Espaciais (INPE)
Caixa Postal 515 – 12.245-970 – São José dos Campos – SP – Brazil

[2]Polo Regional do Vale do Ribeira – Agência Paulista de Tecnologia do Agronegócio (APTA)
São Paulo, Brazil

[3]Divisão de Satélites e Sistemas Ambientais – Centro de Previsão de Tempo e Estudos Climáticos / Instituto Nacional de Pesquisas Espaciais (CPTEC/INPE)
São Paulo, Brazil

[4]Divisão de Processamento de Imagens – Instituto Nacional de Pesquisas Espaciais (INPE)
Caixa Postal 515 – 12.245-970 – São José dos Campos – SP – Brazil

hbendini@dsr.inpe.br, wilson_moraes@uol.com.br, simone@cptec.inpe.br, eymar@dpi.inpe.br, tkorting@dpi.inpe.br, leila@dpi.inpe.br

Abstract

Banana crops have been affected by a disease known as black sigatoka. There are still few studies that consider the spatial distribution and dynamics involved in the dispersion process, as well as monitoring systems that incorporate such information. In this context, the geospatial tools are promising, especially when integrated to geographic information systems with the capacity to incorporate dynamic data, from different sources, allowing real-time monitoring systems. Thus, this ongoing paper proposes the implementation of a system of monitoring and alerting for black sigatoka with the characteristics mentioned, using TerraMA^2 platform.

Resumo

A cultura da bananeira vem sendo assolada por uma doença conhecida por sigatoka-negra. Ainda são poucos os estudos que considerem a sua distribuição espacial e dinâmica envolvida no processo de dispersão, tampouco sistemas de monitoramento que integrem essas informações. Neste contexto, as geotecnologias apresentam ferramentas promissoras, sobretudo quando integradas a sistemas de informações geográficas capazes de incorporar dados dinâmicos, de diversas fontes, permitindo o monitoramento em tempo real. Assim este trabalho, em andamento, propõe a implementação de um sistema de monitoramento e alerta para a sigatoka-negra com as características mencionadas, utilizando a plataforma TerraMA^2.

1. Introdução

A cultura da bananeira vem sendo assolada pela ocorrência da sigatoka-negra. A principal prática utilizada pelos produtores é o controle químico, integrado a técnicas de monitoramento e manejo (GASPAROTTO et al., 2006; MORAES et al., 2006). Ainda são incipientes estudos que considerem a distribuição espacial da doença, bem como a dinâmica envolvida nesse processo de dispersão, e o mapeamento de áreas mais ou menos favoráveis a sua ocorrência. O desenvolvimento de mapas de risco acoplados a modelos de predição pode ser útil para indicar áreas ou épocas mais favoráveis à epidemias. Com uma base de dados ampla, sistemas computacionais podem ser utilizados para verificar se as condições meteorológicas estão favoráveis à infecção pelos patógenos. O geoprocessamento apresenta-se como uma ferramenta viável e promissora para modelar a predisposição à ocorrência da sigatoka-negra no Brasil com base nas condições meteorológicas favoráveis ao seu desenvolvimento (BENDINI *et al.*, 2013).

Aplicações modernas de SIG (Sistema de Informação Geográfica) seguem a tendência impostas pelos avanços nos sistemas de informática, como acesso web, segurança, capacidade computacional, armazenamento, entre outros, demandando cada vez mais carga computacional, gerando grandes bancos de informação. Neste contexto, as aplicações baseadas no modelo cliente-servidor apresentam diversos inconvenientes, tais como, sobrecarga no servidor, falta de robustez no processamento, custos de aquisição e manutenção, além de não contemplar plenamente os requisitos demandados pelos sistemas de emergência e alerta (MOMO et al., 2011). Diante das necessidades de incorporar dinâmica ao sistema, com vistas a permitir um monitoramento em tempo real, isentar o usuário de etapas de pré-processamento e armazenamento dos dados de entrada do modelo, indica-se como solução a utilização de uma arquitetura orientada a serviços, com interoperabilidade via webservices. A plataforma TerraMA^2 (INPE, 2012) é um software livre, baseado em uma arquitetura de serviços que possibilita o desenvolvimento de sistemas operacionais para monitoramento de alertas a riscos ambientais. É um sistema baseado na Arquitetura Orientada a Serviços que possibilita a integração de dados de diversas fontes, permitindo o estabelecimento de cenários de alerta através de análises previamente estabelecidas em linguagem de programação LUA, emitindo ainda notificações de alerta aos usuários. A partir destas considerações, a presente proposta trata da implementação de um sistema de monitoramento e alerta para a sigatoka-negra utilizando a TerraMA^2.

2. Favorabilidade à sigatoka-negra

O desenvolvimento da sigatoka-negra é favorecido por temperaturas médias mensais entre 20 °C e 30 °C, umidade relativa acima de 70 % e precipitação acumulada superior a 100 mm (JACOME & SCHUH, 1992; MOULIOM-PEFOURA et al., 1996; ROMERO & SUTTON, 1997; FUKUDA & MORAES, 2007). A Tabela 1 apresenta os limiares de favorabilidade à doença, para cada variável. Deste modo, é possível inferir sobre a favorabilidade à doença, considerando a álgebra booleana, por meio da Equação 1.

$$\begin{cases} \text{SE } (Tmed = favorável) \text{ AND } (Pp = favorável) \text{ AND } (UR = favorável) \rightarrow \boldsymbol{Favorável} \\ \text{CASO CONTRÁRIO} \rightarrow \boldsymbol{Desfavorável} \end{cases} \quad \text{[Eq. 1]}$$

Tabela 1. Intervalos dos limiares de favorabilidade à doença.

	Favorável	Desfavorável
Precipitação acumulada (*Pp*)	≥100 mm	<100 mm
Temperatura média (*Tmed*)	≥20 ºC e ≤30 ºC	<20 ºC e >30 ºC
Umidade relativa do ar (*UR*)	≥70%	<70%

Moraes et al. (2006) analisaram o progresso da sigatoka-negra durante o período de fevereiro a dezembro de 2005, na região do Vale do Ribeira, onde simularam o comportamento da doença com uma função de regressão, em que as médias de temperatura máxima e mínima e a precipitação acumulada durante uma e duas semanas antes da leitura foram os parâmetros de entrada, tendo sido os que mais se correlacionaram (r=0,82) com a severidade da doença com duas semanas de antecedência, obtendo a Equação 2.

$$\mathrm{EE} = -1265 + 0{,}5886\mathrm{pp} + 73{,}7879\mathrm{tmax} + 52{,}8995\mathrm{tmin} \quad \text{[Eq. 2]}$$

3. Resultados preliminares

Foram realizados estudos preliminares em que a metodologia consistiu de uma etapa de aquisição de dados nas bases de dados meteorológicos, contemplando dados de modelos e de estimativas por satélite, onde também foi necessária uma etapa de processamento destes dados para serem importados para plataforma de um SIG, onde então foram implementados os métodos de geoprocessamento para análise e visualização dos mapas. Foram duas abordagens de análise. A primeira baseou-se em inferência booleana, considerando a Equação 1, com os limiares da Tabela 1, para todo o território nacional, com intervalos mensais. A segunda abordagem considerou apenas o estado de São Paulo, utilizando o modelo da Equação 2, com intervalo quinzenal. As variáveis meteorológicas consideradas no estudo foram precipitação acumulada (Pp) quinzenal e mensal, temperatura máxima média (Tmax) quinzenal e mensal, temperatura mínima média (Tmin) quinzenal e mensal, temperatura média (Tmed) mensal e umidade relativa média mensal (Ur). Para as variáveis temperatura média mensal, e umidade relativa mensal, foram utilizados os dados do modelo ETA (disponível em http://dadosclima.ccst.inpe.br/), na resolução de 40 km. Os dados de temperatura média são fornecidos em Kelvin em arquivos com extensão NetCDF (*Network Common Data Form*). Para obtenção dos dados de precipitação acumulada mensal e quinzenal, foram utilizados os dados do produto 3B42, do satélite TRMM (*Tropical Rainfall Measuring Mission*), disponibilizados com frequência diária na base de dados "Giovanni" (GES DISC *Interactive Online Visualization and analysis Infrastructure*), da NASA *Goddard Earth Sciences Data and Information Services Center* (GES DISC) (disponível em http://gdata1.sci.gsfc.nasa.gov/). Este produto utiliza estimativas de precipitação por micro-ondas do imageador de microoondas (TMI) corrigidas por informações da estrutura vertical das nuvens obtidas do radar de precipitação (PR). Os dados foram fornecidos em $mm.h^{-1}$ em arquivos raster com extensão HDF (*Hierarchical Data Format*) e resolução espacial de 0,25º (cerca de 30 km). Para os dados diários de Tmax e Tmin foi utilizada a base de dados NCEP *Climate Forecast System Reanalysis* (CFSR) (disponível em http://www.esrl.noaa.gov/), na mesma resolução de 0,25º aproximadamente. Obtidos os dados, foram formatados os arquivos ASCII para importação para o Sistema de Processamento de Informações Geográficas (SPRING)

(versão 5.2.1) (CÂMARA et al., 1996), que foi utilizado como plataforma para desenvolvimento do trabalho. Após a importação do dado tipo amostra, foi realizada uma interpolação por média ponderada para geração das grades regulares, com 40 km de resolução. Com as grades regulares, efetuou-se a operação de fatiamento, considerando os limiares estabelecidos na literatura (Tabela 1). Assim, por meio da Equação 1 foram gerados os mapas de favorabilidade. A Figura 1 ilustra os mapas de favorabilidade obtidos para os meses de janeiro a dezembro de 2010.

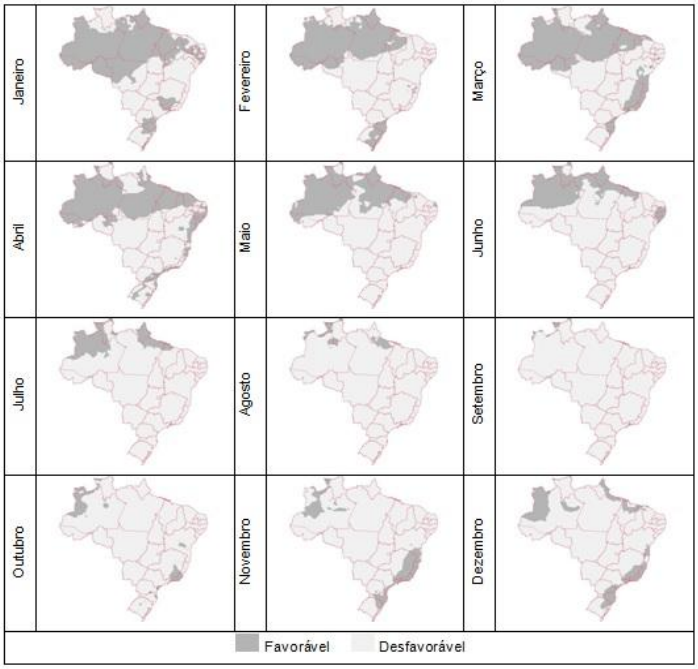


Figura 1. Mapa de favorabilidade climática à sigatoka-negra da bananeira no Brasil, para os meses de janeiro a dezembro, no período de 2010.

Finalmente, para elaboração dos mapas de severidade por meio da Equação 2, foram utilizados os dados diários de Tmax, Tmin, e Pp, sendo que estes foram calculados para períodos quinzenais, para entrada no modelo, devido ao fato deste ter sido ajustado com maior correlação para este intervalo. Para validação destes mapas, de forma a considerar a variabilidade temporal, foram utilizados dados de estado da evolução (EE), obtidos por meio do monitoramento pela metodologia de Fouré (1988), em duas propriedades produtoras de banana nas regiões de Pariquera-açu e Paranapanema, no estado de São Paulo. Foi realizado um estudo de correlação entre os valores obtidos pelo modelo, e os valores obtidos por medidas de campo, onde os coeficientes de correlação foram 0.445 e 0.668, para Paranapanema e Pariquera-açu respectivamente. Como considerações destes resultados preliminares, evidencia-se que tal abordagem pode ser útil para desenvolvimento de plataformas para realização de trabalhos sobre a distribuição espaço-temporal de doenças agropecuárias. Além de contribuir para com a interoperabilidade com dados meteorológicos de estimativas por satélite e modelos numéricos. Evidentemente são necessários estudos com séries temporais maiores e mais informações de campo para validação, a fim de se obter resultados mais consistentes. Para isso, é imprescindível a utilização de sistemas que permitam uma maior dinâmica na entrada de dados.

4. Proposta de Sistema de monitoramento da sigatoka-negra baseado em variáveis ambientais utilizando a TerraMA^2

Tendo como aporte ilustrativo a Figura 3, a proposta em andamento é definida por estabelecer inicialmente um banco de estudos, onde será realizada a validação da metodologia, e uma pré-análise dos dados meteorológicos obtidos por modelos de previsão, para serem associadas incertezas às medidas para a região analisada. E um banco de operação, para geração e visualização dos alertas de modo operacional.

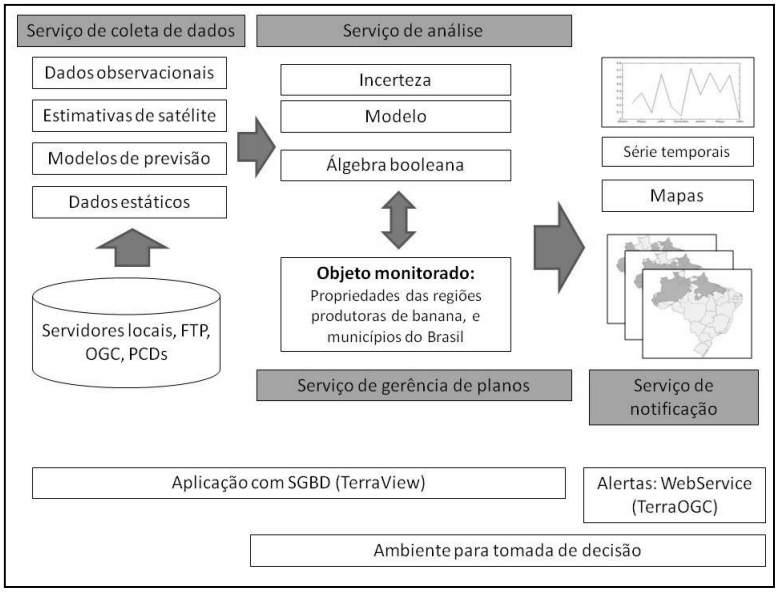


Figura 3. Arquitetura da proposta do sistema de monitoramento da sigatoka-negra baseado em variáveis ambientais utilizando a TerraMA^2.

Os dados necessários para a operação da plataforma incluem dados ambientais, sendo estes os dados dinâmicos coletados automaticamente em servidores FTP, ou disponibilizados para clientes HTTP, seguindo a especificação da OGC (*Open Geospatial Consortium*), ou ainda por Plataformas de Coleta Digitais (PCDs), e que informam sobre a condição das variáveis obtidas a intervalos de tempo pré-determinados, e os objetos monitorados, que incluem dados com informações sobre as pré-condições necessárias para a ocorrência da doença. A princípio, serão propostas as duas abordagens de análise citadas anteriormente. As análises serão baseadas em objeto monitorado, onde os objetos serão os municípios brasileiros na primeira abordagem, e as propriedades de bananicultura de regiões produtoras, considerando a segunda. Para a primeira abordagem serão considerados os limiares estabelecidos na Tabela 1. Já no segundo caso, será utilizado o modelo da Equação 2. As análises serão desenvolvidas com a linguagem de programação LUA (IERUSALIMSCHY et al., 1993). Finalmente, os usuários serão produtores rurais, e tomadores de decisão em geral, que poderão visualizar os alertas por uma aplicação web, conectada ao banco de dados, capaz de apresentar as camadas associadas à análise, seus dados, histórico de alertas e metadados, havendo ainda um serviço de notificação, onde os usuários cadastrados receberão relatórios de alteração, enviados por correio eletrônico.

5. Referências

Bendini, H. N., Moraes, W. S., Silva, S. H. M. G., Tezuka, E. S., Cruvinel, P. E. (2013). "Análise de risco da ocorrência de sigatoka-negra baseada em modelos polinomiais: um estudo de caso". Tropical Plant Pathology 38(1): p. 35 – 43.

Câmara G., Souza R. C. M., Freitas U. M., Garrido J. C. P. (1996) "SPRING: Integrating remote sensing and GIS with object-oriented data modeling". Computers and Graphics 6: p. 13-22.

Fouré E. (1988) "Stratégies de lutte contre la cercosporiose noire des bananiers et des plantains provoquée par *Mycosphaerella fijiensis* Morelet". L'avertissement biologique au Cameroun. Evaluation des possibilités d'amélioration. Fruits 43:269-274.

Fukuda E., Moraes W. S. (2007) "Monitoramento da severidade da Sigatoka Negra em bananais do Vale do Ribeira, São Paulo, Brasil". In: Congresso de Iniciação Científica da UNESP, Resumos... Ilha Solteira SP. UNESP. p. 22 – 23.

Gasparotto, L., Pereira, J. C. R., Hanada, R. E., Montarroyos, A. V. V. (2006) "Sigatoka-negra da bananeira". Manaus: Embrapa Amazônia Ocidental, 177p.

Ierusalimschy, R., Celes, W., Figueiredo, L. H. and Souza, R. (1993) "Lua: uma linguagem para customização de aplicações", In: VII Simpósio Brasileiro de Engenharia de Software, page 55. Caderno de Ferramentas.

INPE – Instituto Nacional de Pesquisas Espaciais. Terra Monitoramento Análise e Alerta – TerraMA^2. 2012. Disponível em: <http://www.dpi.inpe.br/terrama2/>. Acesso em: out. 2014.

Jacome, L. H., Schuh, W. (1992) "Effects of leaf wetness duration and temperature of development of black Sigatoka disease on banana infected by *Mycosphaerella fijiensis* var. difformis", p. 515 – 520. *Phytopathology*. v. 82.

Momo, M. R., Refosco, J. C. (2011) "Arquitetura computacional baseada em computação GRID, aplicada a sistemas de informação geográfica na gestão de risco e alerta da bacia do rio Itajaí", In: Simpósio Brasileiro de Sensoriamento Remoto, Anais... Curitiba PR. INPE. p. 8865.

Moraes W. S., Fukuda E., Mendonça J. C., Silva C. M., Silva S. H. M. (2006) "Behaviour of black Sigatoka in banana plantations of the Ribeira Valley, São Paulo, Brazil", In: XVII Reunião Internacional da Associação para a Cooperação em Pesquisas sobre Banana no Caribe e América Tropical (ACORBAT), Resumos... Joinville SC. IFAC. p. 656 – 661.

Mouliom-Pefoura, A., Lassoudière, A., Foko, J., Fontem, D. A. (1996) "Comparison of development of *Mycosphaerella fijiensis* and *Mycosphaerella musicola* on banana and plantain in various ecological zones in Cameroon". Plant Disease 80: p. 950 – 954.

Romero, R. A., Sutton, T. B. (1997) "Reaction of four Musa genotypes at three temperatures to isolates of *Mycosphaerella fijiensis* from different geographical regions". *Plant Disease* 81: p. 1139 – 1142.

Vistradas: Visual Analytics for Urban Trajectory Data

Luciano Barbosa[1], **Matthías Kormáksson**[1], **Marcos R. Vieira**[1], **Rafael L. Tavares**[1,2], **Bianca Zadrozny**[1]

[1]IBM Research – Brazil

[2]Univ. Federal do Rio de Janeiro – Brazil

{lucianoa,matkorm,mvieira,rafaelt,biancaz}@br.ibm.com

Abstract. *In the past few years a growing number of cities have started monitoring the position of public transportation vehicles using GPS devices. Most of these trajectory data are released in raw format and usually have issues, such as measurement errors. Providing insights from these valuable (and noisy) data is a major challenge in larger cities. In this paper we present a system, called Vistradas, for visual analytics of urban trajectory data. Vistradas allows users to analyze use cases related to trajectories of public buses such as: analysis of bus uniformity, verification of bus route, and the impact of events in bus traffic. Our proposed Vistradas system helps user to get insights into various aspects of public transportation.*

1. Introduction

There is a fast growing use of new technologies (e.g., cheap GPS devices, ubiquitous sensor and cellular networks) that generate large amount of data in the form of trajectories. Basically, a trajectory is a sequence of pairs (location, timestamp), which may contain some other attributes (e.g., temperature, velocity), generated by a moving object. Since trajectory data in their raw format do not bring much valuable information to users, data analytics have a key role to help people get useful insights from trajectory data.

This paper presents an on-going research project for visual analytics of urban trajectory data. Vistradas is a Web-based visualization system that allows users to visualize use cases related to trajectories of public vehicles. In this paper, we make use of trajectories of public buses in the city of Rio de Janeiro. As the data presented noisy information, we implemented a pre-processing step to deal with these issues, and also a data normalization step for our use cases (see Section 3). For the processed data, we created use cases (described in Section 4) that focused on quality of bus service, namely: **(1)** analysis of bus uniformity, **(2)** verification of bus route, and **(3)** the impact of events in bus traffic. Data reports from these use cases can help city bus authority to inspect and monitor bus services in the city. Finally, in Section 5, we briefly describe the currently research we are conducting with the Vistradas system.

2. Related Work

There are several research and commercial systems that provide features to monitor and visualize trajectory data. For instance, [Pu et al. 2013] uses taxi trajectory data to build visual reports for monitoring city traffic. AITVS [Lu et al. 2006] is a visualization system to analyze, monitor and report traffic conditions. CubeView [Shekhar et al. 2002] is a

Web-based visualization package for building summarizations of traffic trends on top of a multi-dimensional data warehouse. [Albuquerque et al. 2013] describes a system to monitor truck fleets, which can be integrated with tweet data to georeference traffic-related facts. SeMiTri [Yan et al. 2010] is a system that semantically enriches trajectories (i.e., sequence of places where a trajectory has passed/stayed) by using other geographic data sources.

More related to our proposed system are CommonGIS [IAIS 2014] and M-Atlas [M-Atlas 2014]. CommonGIS is a general GIS tool with some capabilities for cleaning, integrating and reporting basic summary statistics of trajectory data. M-Atlas provides mechanisms to store and query trajectory data (e.g., range, nearest neighbor queries with a temporal/relational predicate), as well as features for mining trajectory patterns. Examples of such patterns include finding frequent pattern of movement, finding dense areas of traffic jams, among others.

Nevertheless, our proposed Vistradas system is different from previous systems since we are not only building visualizations/reports for traffic monitoring or a tool to store and query trajectories. Instead, we are interested in providing tools for cleaning, managing, integrating, and analyzing statistically large urban trajectory data in order to provide city insights to public managers. To the best of our knowledge, such analysts is not facilitated in previous works. In particular, we are not aware of any previous systems that analyze the use cases in this paper.

3. Data Analysis

In order to describe the uses cases we first characterize the GPS data obtained from buses operating in the city of Rio de Janeiro. The raw trajectory data was obtained from September 26, 2013 to January 9, 2014. It contains information for more than 9,000 buses of around 400 bus lines in Rio de Janeiro[1]. In total there are more than 100 million GPS entries for the mentioned period.

Each GPS data entry has the location of a bus (latitude and longitude), timestamp, bus ID, line ID, and bus velocity. The time between consecutive GPS measurements ranges from anywhere under a minute to over 10 minutes, with an average of 4 minutes. We also had access to GTFS data, which contain general information about the bus routes, such as bus stop locations and expected schedules. In general each route consists of two trips, one going from origin to destination and the second representing the return trip. The GTFS data contain a complete definition of each such trip as a sequence of line segments tracing the streets of the route from origin to destination.

The raw trajectory data itself presented problems, such as: no information about the direction in which the bus is traveling; and, in some cases, wrong latitude/longitude positions and poor time resolution (i.e., much higher than 10 minutes). To deal with these issues and also to normalize the data, we implemented a pre-processing component, depicted in Figure 1. In the following, we present the main steps of this process.

Cleaning: the first step in the pre-processing phase is, for each bus trajectory tr_i in the dataset, to remove GPS entries with distance higher than δ_s (e.g., δ_s=100 meters) from expected route, and trajectories with GPS resolution higher than δ_t (e.g., δ_t=10 minutes);

[1]The GPS data can be obtained in `www.rio.rj.gov.br/web/dadosabertos`.

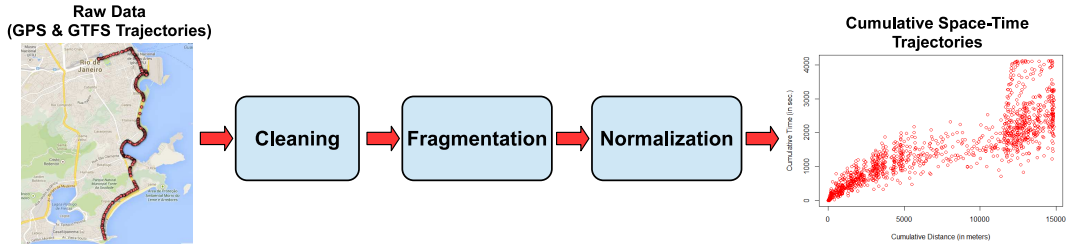


Figure 1. Data pre-processing workflow: cleaning, fragmentation and normalization of raw trajectory data to build cumulative space-time bus movements.

Fragmentation: from the cleaned trajectory dataset, the next step is to discover the bus direction for a given period of time. We fragment tr_i where each fragment represents the bus trip in one of the line directions. To perform these tasks we rely on the expected routes of the line directions using the GTFS data;

Normalization: the final step is to normalize the segmented bus trajectories, which are going to be used in the use cases. This step calculates the cumulative time and distance from the beginning of the bus route for a given bus trip (see examples of cumulative space-time trajectories in Figure 1). To calculate the cumulative distance, we measure the Euclidean distance between consecutive measurements in the expected route provided in the GTFS data (high resolution data). Then, we project a given GPS coordinate on its expected route and calculate its distance to the previous GPS coordinate in the trajectory. Since it is very rare (from our data) to have GPS measurements at the beginning of the bus trips t_0, we calculate t_0 by interpolating the time using the last reported GPS measurement before t_0 and immediately after t_0. Once t_0 is computed, all the other times relatively to t_0 can be easily calculated along the bus route.

4. Use Cases Description

We now describe three use cases our Vistradas system supports using the normalized trajectory data we previously described. These use cases are all related to quality of service of bus lines in the city of Rio de Janeiro.

4.1. Analysis of Bus Uniformity

The first use case is *bus bunching*, a common problem that occurs when two or more buses of the same line are very close to each other along their routes. To detect this problem, for each line we compute a *bus_bunching_score* defined by Equation 1. This score measures how much the spatial distribution of buses along their routes deviates from an expected distribution. A bus line with higher *bus_bunching_score* is more likely to suffer from the bus bunching problem.

The steps to calculate *bus_bunching_score* are: **(1)** for a given bus line, we compute the expected distance between buses by dividing the total number of buses n running on a given time interval (e.g., 4 hours) by the route length l_r; **(2)** then, for all adjacent pairs $n-1$, we calculate how much buses deviate from the expected distance by subtracting it from the observed distance d_i of each pair of adjacent buses (on route) in a given line; **(3)** the final bus bunching score of a line is then the average deviation.

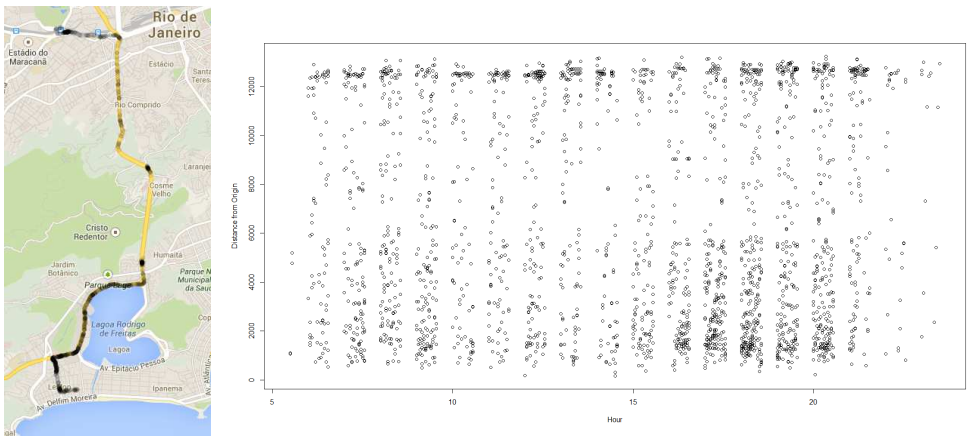


Figure 2. Space-time distribution of buses for line 460 on Nov. 13, 2013.

$$bus\_bunching\_score = \frac{1}{n-1}\sum_{i=1}^{n-1}(d_i - \frac{n}{l})^2 \quad (1)$$

For the *bus bunching* analysis, we first rank the buses according to the *bus_bunching_score* value during a particular day. Then, given a ranked bus line, we show a map with the corresponding distribution of buses and a graph presenting the distribution of the buses in time and space. For this visualization, each dot on the map (and on the graphic) represents a single bus. For example, Figure 2 shows a high bus bunching score for the distribution of buses in line 460 on November 13, 2013. From this example we can observe that buses were very close to each other during rush hours (between 4pm and 8pm) near 4km of their origin route point (at Rodrigo de Freitas Lagoon).

4.2. Verification of Bus Route

The second use case involves the verification of whether buses are respecting their expected routes. This can be useful, for instance, for contract auditors interested in knowing whether the bus companies are respecting their routes, or even for bus companies ensuring their drivers are respecting the expected routes.

Our route verification algorithm works as follows: **(1)** given the expected route r (obtained from the GTFS data) for a line l_r, and the bus GPS measurement p, we calculate the shortest Euclidean distance $min\_dist(p, r)$ between every p for a particular bus in l_r and the sequence of lines defining r; **(2)** in the second step, we measure how much all buses in l_r deviate from the expected route r. We calculate the deviation score *deviation_score* by taking the average $min\_dist(p, r)$ over all GPS measurements of line l_r, as:

$$deviation\_score = \frac{1}{n}\sum_{i=1}^{n} min\_dist(p_i, r) \quad (2)$$

Using Vistradas we are able to detect bus lines with the highest deviations, which can help us understand the source of the problem. In Figure 3, we show an example of bus line 906 that obtained a high deviation value. We can clearly see the buses for line

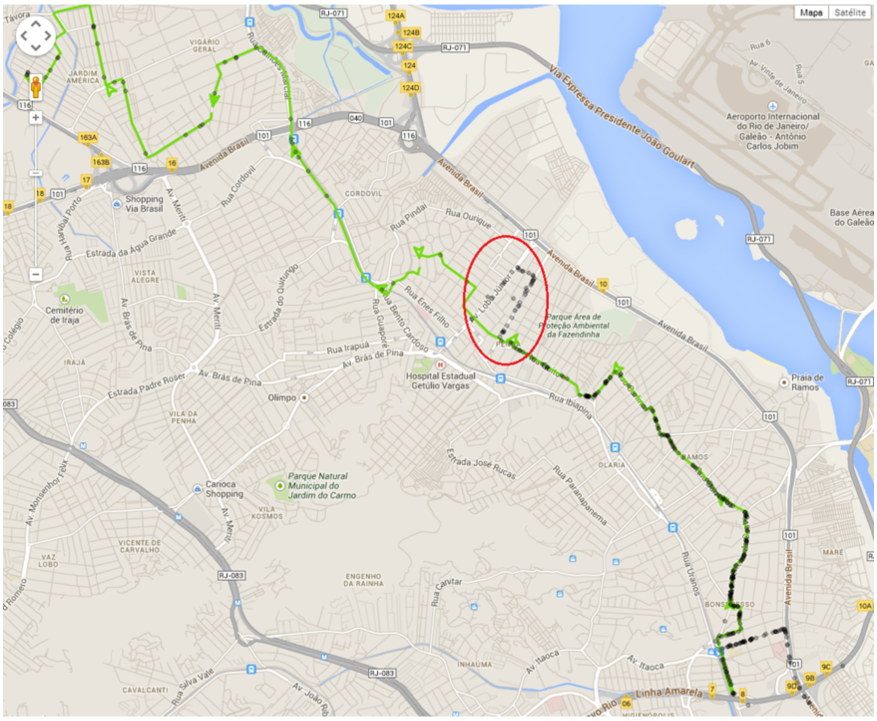


Figure 3. Route verification using the bus line 906. Green line is the expected route provided in the GTFS data, and black dots are the GPS measurements.

906 systematically changed their route on this particular day. We can also observe both the expected route (green line) and the actual route (black dots) on the same visualization for a given line, where there are many GPS measurements (black dots) far away from their expected route (green line). Since the GPS used in the buses are not so accurate in reporting their real locations, the GPS data may contain imprecise measurements, which may be due to the nature of device or occlusion in the area. Nevertheless, we can see many black dots very close to the expected route and a few ones (marked with a red circle) far away from the expected route, which may indicate a possible detour.

4.3. Impact of Events in Bus Traffic

The third use case involves the impact of events on bus traffic. For example, road constructions, natural phenomena (e.g., heavy rain, mudslide), sport or music events, among others, are some events that can have a negative impact on the city's traffic. In order to measure the impact of events, we calculate the difference between the average traffic velocity on the bus routes in a period before and after a given event. Vistradas allows users to select a particular date and then plot the difference of velocities before and after the given date. Figure 4 shows the impact of the Perimetral overpass fall on bus route 121. The region in red color shows a significant impact on bus velocity: a decrease of 8Km/h. We also present the information in a line graphic, so that the user can compare the difference easily and, if desired, select a point in the line to see on the map. This kind of information can be very useful for city planners to verify the impact of planned and unplanned events on city's traffic.

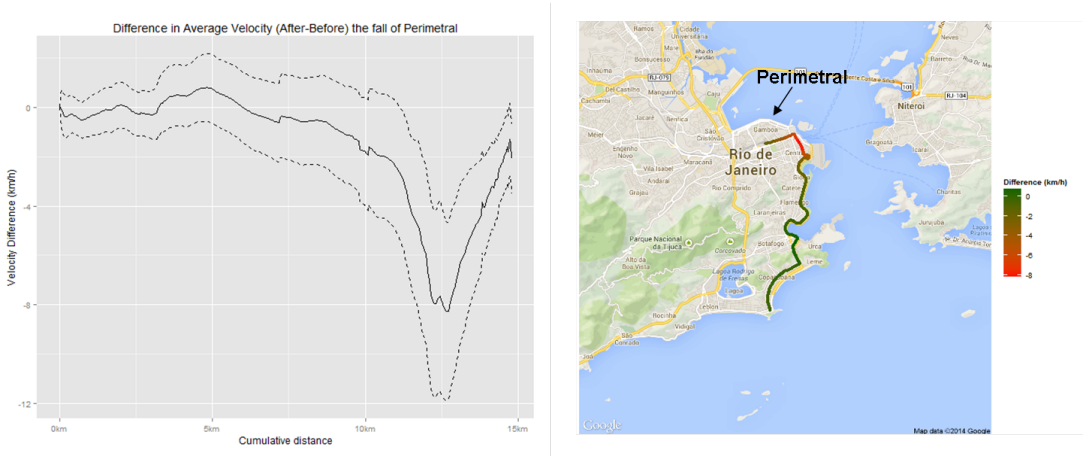


Figure 4. The traffic impact before and after the implosion of Perimetral overpass.

5. Conclusion and Ongoing Research

This paper describes Vistradas, a Web-based system that provides visual analytics of urban trajectory data. This paper shows a few use cases supported by our Vistradas system using real bus GPS data.

As for the current status of our Vistradas system, we are developing two new use cases for real problems that the city of Rio de Janeiro faces: **(1)** we are integrating the bus GPS data with data coming from weather stations, microbloging, and news websites to get new insights of the city; and **(2)** using the historical GPS data, we are building predictive models to make time-prediction of buses [Kormaksson et al. 2014] for a few particular scenarios (e.g., when will the bus line 121 arrive at my stop 15? when will I get to my final destination?).

References

Albuquerque, F., Casanova, M., Macêdo, J., Carvalho, M., and Renso, C. (2013). A proactive application to monitor truck fleets. In *Proc. of IEEE MDM*, pages 301–304.

IAIS, F. (2014). CommonGIS. www.iais.fraunhofer.de/1871.html?&L=1.

Kormaksson, M., Barbosa, L., Vieira, M., and Zadrozny, B. (2014). Bus travel time predictions using additive models. In *Proc. of IEEE ICDM*.

Lu, C.-T., Boedihardjo, A. P., and Zheng, J. (2006). Aitvs: Advanced interactive traffic visualization system. In *Proc. of IEEE ICDE*, pages 167–167.

M-Atlas (2014). M-Atlas. http://www.m-atlas.eu/.

Pu, J., Liu, S., Ding, Y., Qu, H., and Ni, L. M. (2013). T-watcher: A new visual analytic system for effective traffic surveillance. In *Proc. of IEEE MDM*, pages 127–136.

Shekhar, S., Lu, C. T., Liu, R., and Zhou, C. (2002). Cubeview: a system for traffic data visualization. In *Proc. of IEEE Int'l. Transp. Sys.*, pages 674–678.

Yan, Z., Spremic, L., Chakraborty, D., Parent, C., Spaccapietra, S., and Aberer, K. (2010). Automatic construction and multi-level visualization of semantic trajectories. In *Proc. of ACM SIGSPATIAL*, pages 524–525.

Mapeamento Participativo de Opiniões sobre o Uso de Dinheiro Público

Henrique Ferreira Soares[1], **Michele Brito Pinheiro**[1], **Clodoveu A. Davis Jr.**[1]

[1] Depto. de Ciência da Computação – Universidade Federal de Minas Gerais (UFMG)
Av. Pres. Antônio Carlos 6.627 – 31.270-210 – Belo Horizonte – MG – Brasil

{hsoares,mibrito,clodoveu}@dcc.ufmg.br

Abstract. *Recent surveys show dissatisfaction of the population with politicians and their representatives. This paper describes a technological tool developed to provide a communication channel between citizens and policy makers. This tool allows to make reports like the lack of investment and waste of public resources, and highlight initiatives which has society approval. The reports will be georeferenced and stored in a database, where they can be seen and evaluated by other users.*

Resumo. *Pesquisas recentes apontam a insatisfação da população com os políticos que a representam. Este trabalho descreve uma ferramenta tecnológica desenvolvida para prover um canal de comunicação entre os cidadãos e os gestores públicos. A ferramenta permite fazer denúncias como a carência de investimento e desperdício de recursos públicos, e destacar iniciativas que obtiveram aprovação da sociedade. As denúncias serão georreferenciadas e armazenadas em um banco de dados, onde elas poderão ser vistas e avaliadas por outros usuários.*

1. Introdução

Após a indicação do Brasil como país sede da Copa do Mundo de 2014, e principalmente, durante o período da Copa das Confederações, diversos movimentos se organizaram, deflagrando em manifestações por todo o país. Dentre os motivos da insatisfação da população estavam os gastos de dinheiro público em obras para o suporte do evento, ao invés do investimento em setores mais necessitados que compõem os serviços básicos.

Diante desse cenário, o presente artigo propõe uma ferramenta que permite que o cidadão opine sobre o uso de recursos públicos distribuídos espacialmente. A aplicação, construída como parte de um projeto de pesquisa que visa gerar um *framework* de desenvolvimento de sistemas de coleta voluntária de dados geográficos, recebe contribuições de qualquer cidadão, permitindo que esses indiquem pontos de seu conhecimento e expressem suas opiniões, podendo ser favoráveis ou contrárias aos gastos orçamentários. Sendo assim, o conjunto das contribuições oferece às autoridades uma melhor percepção acerca da visão da população sobre seu desempenho como gestores, e destaca as iniciativas que apresentam aprovação da sociedade. Ademais, a ferramenta dá suporte ao debate direto entre os cidadãos, que podem interagir através de comentários e ao avaliar contribuições de outros cidadãos. Do ponto de vista científico, a ferramenta constitui um protótipo para projeto e avaliação dos recursos necessários para aplicações interativas para coleta

de dados geográficos, envolvendo aspectos tais como identificação, acompanhamento das contribuições, validação das contribuições por pares, *feedback* e outros.

Este artigo é organizado como se segue. A Seção 2 apresenta trabalhos relacionados. Na Seção 3 é caracterizada a arquitetura do aplicativo. A Seção 4 descreve detalhes de implementação e explica o funcionamento da aplicação. Finalmente, a Seção 5 traz conclusões e lista trabalhos futuros.

2. Trabalhos Relacionados

Recentemente, com a evolução da Web para a *Web 2.0*, participativa, e o lançamento da API do Google Maps, deu-se início ao processo de "democratização dos Sistemas de Informação Geográfica" [Butler 2006]. Nesse contexto, diversas aplicações que visam coletar dados geográficos na Web surgiram, como o OpenStreetMap[1], Wikimapia[2], Wikicrimes[3] e Strepitus[4]. Aos dados obtidos por esse tipo de *Crowdsourcing* é atribuído o nome *Informação Voluntaria Geográfica* (*Volunteered Geographic Information*, VGI) [Goodchild 2007], e podem apresentar inúmeros temas de contribuição.

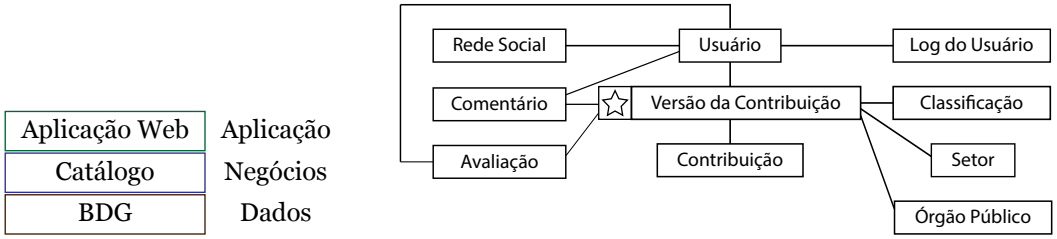
No universo de aplicações VGI, existem vertentes que vão do mapeamento básico à participação pública em ações de governo. A primeira engloba iniciativas que se destinam à criação e correção de bases de dados que contêm camadas correlatas às definidas pelo *framework* de dados introduzido pelo *Federal Geographic Data Committe*[5]. Nessa direção se enquadram o OpenStreetMap, Wikimapia e o Common Census[6]. A outra tendência compreende os *Public Participation Geographic Information Systems* (PPGIS), que incentivam os cidadãos da sociedade a contribuir com dados de seu conhecimento, fornecendo informações relevantes que podem ser utilizadas como forma de expressão popular para influenciar as tomadas de decisão do poder público [Drew 2003], [Elwood 2006]. A exemplo dessas aplicações cabe citar o Ushahidi[7] [Okolloh 2009], Wikicrimes [Furtado et al. 2010] e Strepitus [Vellozo et al. 2013].

A coleta de dados por contribuição voluntária apresenta características importantes, como a velocidade de atualização e baixo custo de implementação, que chamam a atenção quando comparados com métodos de coleta tradicionais [Craglia 2007]. Porém, sua utilização em larga escala pela comunidade científica é limitada por questões como a qualidade e credibilidade [Flanagin and Metzger 2008]. Existem trabalhos dedicados à identificação de mecanismos de garantam essa qualidade [Goodchild and Li 2012]. Outros propõem a utilização de sistemas de reputação que definem papéis aos usuários, sendo possível a validação e publicação de apenas dados relevantes [Maué 2007]. Outra questão que é explorada em pesquisa é a identificação dos motivos que levam usuários a contribuírem e se manterem ativos utilizando essas ferramentas [Coleman et al. 2009]. Por fim, existem ainda trabalhos voltados para o desenvolvimento de *frameworks*, que objetivam a flexibilidade e redução do tempo de desenvolvimento de novas aplicações [Davis Jr et al. 2013][Sheppard 2012][Silva and Davis Jr 2008].

[1] http://www.openstreetmap.org
[2] http://wikimapia.org
[3] http://www.wikicrimes.org/
[4] http://aqui.io/strepitus/
[5] http://www.fgdc.gov/framework/frameworkintroguide/
[6] http://commoncensus.org
[7] http://www.ushahidi.com

3. Arquitetura e Modelagem

O presente trabalho segue o padrão proposto pelo *framework* descrito por [Davis Jr et al. 2013]. Dessa forma, a arquitetura da aplicação se divide em três camadas: *Dados*, *Apresentação* e *Serviços* (Figura 1(a)). A camada de dados pode ser vista também como uma camada de persistência, sendo composta pelo Sistema de Gerenciamento de Banco de Dados (SGBD), que recebe um esquema físico derivado da modelagem conceitual do sistema.



(a) Arquitetura (b) Diagrama de Classes

Figura 1. Arquitetura e Modelagem

O esquema conceitual do *framework* estabelece dez classes, das quais o atual trabalho faz uso de quatro: "Usuário", "Log de Usuário", "Versão da Contribuição" e "Comentários", e estende a classe "Tipo de Contribuição" em três: "Classificação, Setor", "Órgão Público" (Figura 1(b)). "Usuários" é a entidade que mantém dados de identificação, que pode ser derivada do cadastro do usuário em uma rede social, como Facebook ou Twitter. O "Log do Usuário" armazena dados de acesso, como sessões iniciadas e finalizadas pelo usuário. As "Versões da Contribuição" são armazenadas, permitindo que alterações posteriores sejam realizadas pelos contribuintes, além de viabilizar analises sobre o processo de edição dos dados. A classe "Contribuições" representa a compilação das versões mais recentes das colaborações. Por fim, a entidade "Comentários" armazena as opiniões dos usuários sobre cada contribuição.

Para a caracterização das contribuições, o *framework* também previa uma classe chamada "Tipo de Contribuição". No presente trabalho esta classe foi transformada em três outras: "Classificação", "Setor" e "Órgão Público". A "Classificação" identifica o tipo de relato realizado, podendo indicar desperdício de dinheiro público, associadas a lugares onde os cidadãos identificam mau uso dos recursos, ou como carência de investimento, em regiões nas quais o cidadão acha que deveria haver mais investimento, e bom investimento, quando a inciativa foi bem executada e contribuiu com a população. O "Setor" apresenta um catálogo de setores públicos, como Saúde, Educação, Transporte, dentre outros. Finalmente, o "Órgão Responsável" inclui órgãos públicos de diferentes esferas de governo.

A arquitetura ainda prevê a camada de *Negócios*, em que foi utilizado um provedor de serviços Web OGC, para oferecer camadas de dados geográficos compiladas das contribuições, sendo possível que outros trabalhos possam também fazer uso dos dados. A camada de *Aplicação* foi organizada utilizando uma arquitetura em *Model,View,Controller* (MVC), destinada ao ambiente *Web*.

4. Implementação e Aplicação

Como descrito anteriormente, a aplicação, denominada MapaOpinião[8], é composta por três camadas. A camada de *Dados* foi implementada utilizando o SGBD PostgreSQL juntamente com a extensão espacial PostGIS. Tais componentes foram escolhidos levando em consideração sua qualidade e por fazerem de projetos de código aberto com ampla participação da comunidade de desenvolvedores.

Para a camada de Negócios, o GeoServer foi adotado como provedor de serviços por ser capaz de se conectar a diversos SGBDs. Dessa maneira, a aplicação pode ser expandida, permitindo a utilização de várias fontes de dados ao mesmo tempo.

Na camada de apresentação, a interface Web foi desenvolvida para ser interpretada por qualquer navegador. Ela foi construída utilizando Code Igniter, um *framework* MVC escrito em PHP. Foram desenvolvidos quatro *Controlles*: Usuário, Contribuição, Comentário e Avaliação. O primeiro módulo, Usuário, é responsável pelo login e registro de sessão dos usuários no sistema. Esse módulo inclui a funcionalidade não prevista inicialmente no *framework* de VGI, que permite de login por APIs *OAuth* utilizando a biblioteca *HybridAuth*. O segundo módulo, de Contribuição, é responsável pela validação dos tipos de dados fornecidos pelos usuários. Por fim, os módulos de Comentário e Avaliação possuem funcionalidade similares e, respectivamente, tratam as informações coletadas, que são enviadas pelo *Model* para o banco de dados. Para cada *Controller*, foi desenvolvido um respectivo *Model*, que é responsável pelas transações dados utilizadas por estes componentes.

Foi desenvolvida também uma *View* principal, que é utilizada para exibição da interface da aplicação. Esse componente é composto pelo mapa base e uma barra de ferramentas. O mapa base foi desenvolvido utilizando a biblioteca OpenLayers e permite a escolha de uma das três camadas: Google Maps OpenStreetMap e Geoserver. A biblioteca ainda auxilia na identificação de novos pontos que podem ser inseridos pelos usuários no ato da contribuição. A barra de ferramentas inclui funcionalidades como login, caso o usuário não tenha iniciado uma sessão, logout, destinada aos usuários que já estão em uma sessão, e a ativação e desativação da edição do mapa. Além destes componentes, foram desenvolvidos formulários para inserção das informações sobre as contribuições, estes foram desenvolvidos utilizado PHP e JQuery.

A ferramenta disponibiliza os dados coletados sem a necessidade de realizar login. Assim qualquer usuário pode acessar as informações e realizar consultas. Aos usuários registrados, é permitido opinar sobre o conteúdo previamente inserido através de comentários e sob a forma de uma avaliação positiva ou negativa a respeito de uma contribuição, estabelecendo assim uma forma de validação colaborativa do conteúdo. Dessa forma, contribuições que apresentam muitas avaliações negativas podem ser eliminadas e deixam de aparecer no mapa base.

A aplicação apresenta um funcionamento simples, de modo que para realização de uma nova contribuição o usuário deverá inicialmente realizar login, utilizando a conta de uma das redes sociais listadas anteriormente. Em seguida, o usuário deve escolher o local onde deseja inserir a contribuição. Nos passos seguintes ele deve preencher o formulário, fornecendo a classificação da contribuição, o setor e órgão público considerado

[8]http://aqui.io/mapaopiniao

Figura 2. Interface da aplicação

responsável. Além dessas informações, ao usuário é oferecida a possibilidade de inserir uma justificativa para sua contribuição. Como no caso de outros aplicativos VGI, não é feita qualquer crítica ou validação sobre os dados fornecidos pelo usuário. Isso é deixado a cargo de outros usuários que visitem o site, que podem reagir a postagens e comentá-las, indicando opiniões divergentes.

5. Conclusões e Trabalhos Futuros

Com a ferramenta, cidadãos poderão denunciar o mau uso do dinheiro público ou a necessidade de investimentos, apontando a localização no mapa e colocando mais informações pertinentes à denúncia. Apesar de ser voltado para a manifestação de opiniões e impressões pessoais, e portanto sujeito a conflitos de opinião, espera-se que o aplicativo sirva no futuro como um meio de comunicação entre população e o Estado. O conjunto das colaborações permitirão perceber, por exemplo, regiões que concentram obras polêmicas ou de qualidade duvidosa, ou pontos sobre os quais o debate é mais intenso. Em outra direção, pode ser possível indicar lugares que não recebem um volume razoável de investimento público.

Como um trabalho ainda em desenvolvimento, algumas funções serão inseridas como mecanismos para detecção de contribuições inválidas e para gerar novas visualizações, que são uma importante forma de dar retorno ao tempo dispendido pelos cidadãos ao contribuir. Para a execução do primeiro tópico ainda é necessário receber um volume de contribuições que permita a identificação de perfis e padrões de comportamento, de modo a aplicar métodos probabilísticos para a inferência de comportamentos anômalos. O outro elemento pode contemplar recursos visuais que considerem o aspecto temporal da contribuição, e visualizações que reflitam os resultados de análise a respeito de temas específicos. Além disso, está prevista uma expansão do estudo e análise sobre as contribuições propriamente ditas, visando identificar perfis de voluntários, frequência de contribuição, ritmo de incorporação de novos voluntários, e outros problemas ligados à participação dos cidadãos. Nesse sentido, pretende-se incorporar mecanismos de motivação, através do reconhecimento da reputação de usuários pelo uso continuado do sistema. Mais adiante, um desafio consiste em fazer o conteúdo coletado chegar às autoridades responsáveis.

6. Agradecimentos

Os autores agradecem ao CNPq (308678/2012-5) e FAPEMIG (CEX-PPM-00518/13) pelo apoio no desenvolvimento deste projeto.

Referências

Butler, D. (2006). Virtual globes: The web-wide world. *Nature*, 439(7078):776–778.

Coleman, D. J., Georgiadou, Y., and Labonte, J. (2009). Volunteered geographic information: The nature and motivation of produsers. *International Journal of Spatial Data Infrastructures Research*, 4(1):332–358.

Craglia, M. (2007). Volunteered geographic information and spatial data infrastructure: when do parallel lines converge? *Workshop on Volunteered Geographic Information*.

Davis Jr, C. A., Vellozo, H. S., and Pinheiro, M. B. (2013). A framework for web and mobile volunteered geographic information applications. *XIV Brazilian Symposium on GeoInformatics (GeoInfo 2013)*.

Drew, C. H. (2003). Transparency—considerations for PPGIS research and development. *URISA Journal*, 15(1):73–78.

Elwood, S. (2006). Beyond cooptation or resistance: Urban spatial politics, community organizations, and GIS-based spatial narratives. *Annals of the Association of American Geographers*, 96(2):323–341.

Flanagin, A. J. and Metzger, M. J. (2008). The credibility of volunteered geographic information. *GeoJournal*, 72(3-4):137–148.

Furtado, V., Ayres, L., de Oliveira, M., Vasconcelos, E., Caminha, C., D'Orleans, J., and Belchior, M. (2010). Collective intelligence in law enforcement - The Wikicrimes System. *Inf. Sci.*, 180(1):4–17.

Goodchild, M. F. (2007). Citizens as sensors: the world of volunteered geograph. *GeoJournal*, pages 211–221.

Goodchild, M. F. and Li, L. (2012). Assuring the quality of volunteered geographic information. *Spatial Statistics*, 1(0):110–120.

Maué, P. (2007). Reputation as a tool to ensure validity of VGI. *Workshop on Volunteered Geographic Information*.

Okolloh, O. (2009). Ushahidi, or'testimony': Web 2.0 tools for crowdsourcing crisis information. *Participatory learning and action*, 59(1):65–70.

Sheppard, S. A. (2012). wq: A modular framework for collecting, storing, and utilizing experiential VGI. In *Proceedings of the 1st ACM SIGSPATIAL International Workshop on Crowdsourced and Volunteered Geographic Information*, GEOCROWD '12, pages 62–69, New York, NY, USA. ACM.

Silva, J. C. T. and Davis Jr, C. A. (2008). Um framework para coleta e filtragem de dados geográficos fornecidos voltuntáriamente. *X Brazilian Symposium on GeoInformatics (GeoInfo 2008)*.

Vellozo, H. S., Pinheiro, M. B., and Davis Jr, C. A. (2013). Strepitus: um aplicativo para coleta colaborativa de dados sobre ruído urbano. *IV Workshop de Computação Aplicada à Gestão do Meio Ambiente e Recursos Naturais*.

Reconstrução da geometria de itinerários de ônibus a partir de descrições textuais

Diogo Rennó R. Oliveira[1], **Clodoveu A. Davis Jr.**[1]

[1]Departamento de Ciência da Computação – Universidade Federal de Minas Gerais (UFMG)

{renno,clodoveu}@dcc.ufmg.br

Abstract. *Bus routes are fundamental information for users when planning their way through the city. Reading and understanding the routes, which are usually presented as tables with street names, is a difficult task. In this work, we present a method aimed at retrieving the route's geometry based on its table and the city's road network, which allows viewing the route on a map. Using the proposed method, we developed a web application for the city of Belo Horizonte.*

Resumo. *Os itinerários das linhas de ônibus da cidade são fundamentais para o planejamento de rotas pelos usuários de transporte coletivo. Tradicionalmente apresentados em forma tabular, a leitura dos itinerários a partir somente dos nomes de logradouros é difícil. Neste trabalho, apresentamos um método para reconstrução da geometria dos itinerários a partir das tabelas e da malha viária da cidade, permitindo a visualização no mapa. A partir do método proposto, uma aplicação web foi desenvolvida para a cidade de Belo Horizonte.*

1. Introdução

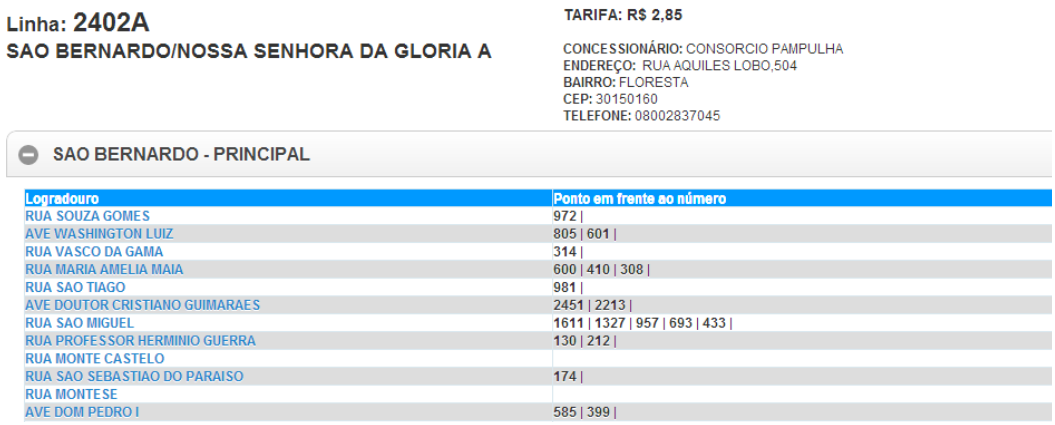
Em Belo Horizonte, o gerenciamento e fiscalização do sistema de transportes e do trânsito são tarefas da BHTRANS (Empresa de Transporte e Trânsito de Belo Horizonte S/A). Entre as atribuições da empresa, está a administração do serviço de transporte coletivo por ônibus, que atende, diariamente, a 1,3 milhões de passageiros na cidade[1]. O sistema abrange mais de 300 linhas, exploradas por quatro consórcios que operam mais de 800 itinerários com mais de 2.800 veículos. As linhas dividem-se em alimentadoras, troncais, interbairros, circulares e radiais. Diante da estagnação do transporte por metrô na cidade, o sistema vem crescendo continuamente, com a criação de corredores exclusivos de ônibus e de um sistema *Bus Rapid Transit* (BRT) em 2014.

O grande número de linhas apresenta uma dificuldade clara ao usuário quando este precisa se locomover para regiões até as quais não conhece o melhor trajeto. Muitas vezes, é necessário utilizar pelo menos duas linhas de ônibus, fazendo o baldeamento entre elas em um ponto adequado do caminho [Cardoso 2007]. Neste contexto, os dois principais serviços disponíveis à população para o planejamento de trajetos são os itinerários da BHTRANS e o Google Maps.

A BHTRANS publica, em sua página na Web[2], os quadros de horários e itineráros das linhas de ônibus. No entanto, as tabelas são de difícil leitura. Em particular, cada itinerário é apresentado como uma lista das ruas e avenidas percorridas pelo ônibus. Além disso, os dados são indexados apenas pelo código das linhas (por exemplo, 2215C, 5102 ou 2211A). A Figura 1 apresenta um exemplo de itinerário disponibilizado. Observe que,

[1]http://bhtrans.pbh.gov.br/portal/page/portal/portalpublico/Temas/Noticias/Passageiros%20%C3%B4nibus%20BH
[2]http://bhtrans.pbh.gov.br

para conseguir planejar um deslocamento, um usuário precisa ter um mapa mental das ruas da cidade, ou identificar paralelamente as ruas em um mapa da cidade.



Linha: 2402A
SAO BERNARDO/NOSSA SENHORA DA GLORIA A

TARIFA: R$ 2,85

CONCESSIONÁRIO: CONSORCIO PAMPULHA
ENDEREÇO: RUA AQUILES LOBO,504
BAIRRO: FLORESTA
CEP: 30150160
TELEFONE: 08002837045

SAO BERNARDO - PRINCIPAL

Logradouro	Ponto em frente ao número
RUA SOUZA GOMES	972 |
AVE WASHINGTON LUIZ	805 | 601 |
RUA VASCO DA GAMA	314 |
RUA MARIA AMELIA MAIA	600 | 410 | 308 |
RUA SAO TIAGO	981 |
AVE DOUTOR CRISTIANO GUIMARAES	2451 | 2213 |
RUA SAO MIGUEL	1611 | 1327 | 957 | 693 | 433 |
RUA PROFESSOR HERMINIO GUERRA	130 | 212 |
RUA MONTE CASTELO	
RUA SAO SEBASTIAO DO PARAISO	174 |
RUA MONTESE	
AVE DOM PEDRO I	585 | 399 |

Figura 1. Um dos itinerários principais da linha 2402A (São Bernardo), como disponibilizado pela BHTRANS.

O Google Maps[3] permite ao usuário pesquisar as melhores opções de trajeto entre dois pontos de Belo Horizonte, com locomoção através do transporte coletivo. O sistema é útil, mas possui algumas limitações. Somente são apresentadas ao usuários as melhores opções computadas (cerca de quatro ou cinco, ordenadas segundo o tempo esperado de deslocamento). Além disso, pode ser interessante para o usuário utilizar alguma linha específica em parte do trajeto, ou percorrer um caminho a pé para pegar apenas um ônibus ou ter mais opções de linhas, mas estas funcionalidades não estão disponíveis. Finalmente, não é possível visualizar o itinerário completo de uma determinada linha, apenas os trechos que fazem parte do trajeto recomendado pelo sistema.

O objetivo deste trabalho é a recuperação da geometria dos itinerários de linhas de ônibus a partir de sua representação textual. Com o método proposto, foi desenvolvido um sistema capaz de apresentar aos usuários, em um mapa da cidade, os itinerários das linhas de ônibus de Belo Horizonte. O usuário deve ser capaz de informar o código da linha e visualizar, no mapa, o trajeto destacado percorrido pelo ônibus, de acordo com o itinerário fornecido pela BHTRANS. A principal contribuição do sistema é viabilizar a consulta simples e rápida dos logradouros por onde determinado ônibus passa, e o trabalho demonstra a viabilidade da aplicação de métodos e técnicas de tratamento de dados textuais associados a dados geográficos como forma de se conseguir a espacialização de sistemas com características semelhantes.

2. Fontes de dados

Duas fontes de dados foram utilizadas no projeto. A primeira consiste nas descrições textuais dos itinerários das linhas de ônibus de Belo Horizonte, coletadas em abril de 2014 a partir do *site* da BHTRANS. Foram coletados todos os itinerários de cada linha (por exemplo, "principal", "retorno no centro da cidade", "domingos e feriados" etc.). Cada itinerário consiste em uma lista de nomes de logradouros, ordenada segundo o trajeto percorrido pelo ônibus. Também foram obtidos o código, tarifa e concessionário de cada linha, além da numeração dos pontos de cada logradouro que possui paradas do ônibus.

[3]http://maps.google.com

A segunda fonte de dados utilizada é a rede de logradouros de Belo Horizonte. A malha viária foi fornecida, em maio de 2014, pela Prefeitura de Belo Horizonte por meio de uma iniciativa de disseminação de dados[4]. Os dados foram importados para um banco de dados do PostgreSQL/PostGIS, e consistem em um conjunto de tabelas contendo nós de rede (cruzamentos entre vias), arcos (trechos de circulação e de conversão) e informações de logradouros (tipo e nome).

Na malha viária, cada logradouro é formado por arcos de circulação. Cada arco de circulação possui geometria e informação do logradouro do qual faz parte. Os arcos são direcionados, e quando o fluxo é bidirecional existe um arco em cada sentido. O cruzamento entre dois logradouros é definido usando um ou mais arcos de conversão, que conectam um trecho de cada logradouro segundo o que é permitido nas regras de circulação de trânsito. Arcos de conversão não fazem parte de nenhum logradouro (possuem identificador de logradouro nulo).

3. Tratamento dos dados

A aplicação do método apresentado na Seção 4 demandou alguns tratamentos das bases de dados. O principal deles, ilustrado no Algoritmo 1, foi a criação da tabela *Trecho*. A tabela contém os atributos convencionais e geográficos de todos os logradouros da base original, de maneira a serem facilmente utilizados pelo método proposto.

Entrada: Tabelas *arcos de circulação* (ACirc) e *arcos de conversão* (AConv)
Saída: Tabela *trecho* (Trecho)

1 $Trecho \leftarrow ACirc$;
2 **repita**
3 $\quad A \leftarrow \{a : a \in AConv, a \notin Trecho, \exists\ e \in Trecho \mid intercepta(a, e)\}$;
4 $\quad Trecho \leftarrow Trecho \cup A$;
5 **até** $|A| = 0$;

Algoritmo 1: *Construção da tabela Trecho*

Inicialmente, todos os arcos de circulação são adicionados à tabela Trecho. Os arcos de conversão são então sucessivamente adicionados. A cada iteração, um arco de conversão a é adicionado à tabela Trecho se a não está em Trecho e se a intercepta algum outro arco de Trecho. Um detalhe importante foi omitido do pseudocódigo: sempre que um arco de conversão a é adicionado à tabela Trecho, ele é associado ao logradouro do arco e, para todo $e \in Trecho$ tal que a intercepta e. O objetivo deste procedimento é associar cada arco de conversão a a todos os logradouros dos quais a intercepta algum arco. A Figura 2 apresenta um exemplo de resultado da aplicação do algoritmo.

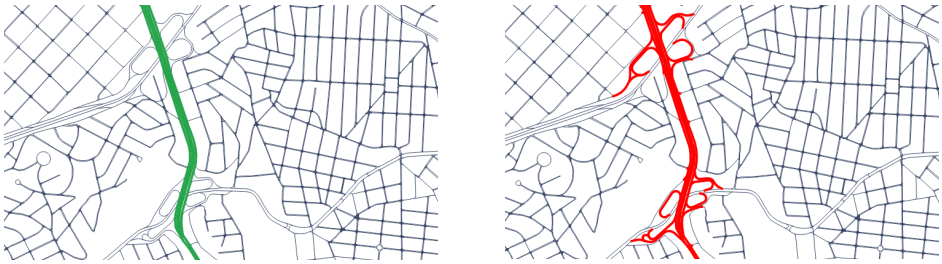


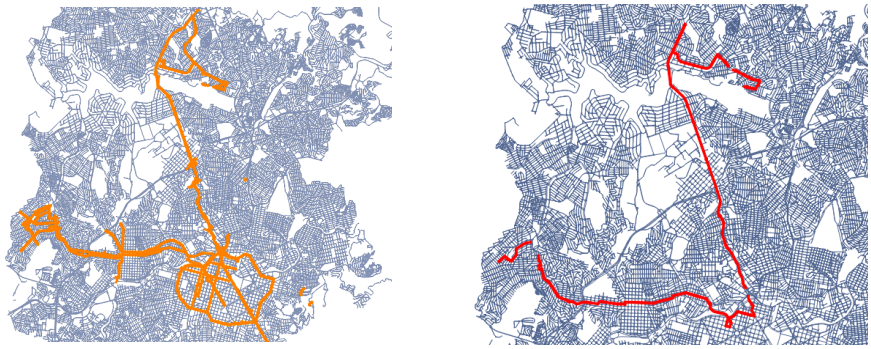
Figura 2. Parte da Avenida Presidente Antônio Carlos recuperada da base original (verde) e da tabela Trecho (vermelho). Nesta última, pode-se observar que os arcos de conversão que conectam a avenida a outros logradouros foram associados a ela.

[4]https://geodadosbh.pbh.gov.br/

4. Método

Foi desenvolvido um método para recuperar a geometria do itinerário das linhas de ônibus, utilizando uma sequência de nomes de logradouros e a geometria dos arcos da malha viária. A única forma de associação entre a base de dados de itinerários de ônibus coletada a partir do *site* da BHTRANS e a malha viária disponibilizada pela prefeitura é o casamento textual dos nomes de logradouros. Felizmente, apesar de cada instituição possuir sua versão dos dados, ambas as bases possuem aparentemente a mesma origem. Todos os 3.694 nomes de logradouros (incluindo o tipo: rua, avenida, praça etc.) presentes nos itinerários foram extraídos e buscados na base da rede viária, e 92% deles puderam ser recuperados com um casamento exato do nome.

A recuperação da geometria completa dos logradouros associados a uma linha não se mostrou um bom método de reconstrução do itinerário para visualização. Como exemplificado na Figura 3 (a), é possível que o ônibus passe por apenas uma parte do logradouro, tornando imprecisa a reconstrução do trajeto a partir da geometria inteira.



(a) **(b)**

Figura 3. (a) Itinerário da linha 2402A construído a partir da geometria completa de todos os logradouros por onde o ônibus trafega; (b) Itinerário da linha 2402A construído a partir dos trechos dos logradouros por onde o ônibus trafega.

O método proposto para recuperação da geometria dos itinerários consiste em determinar o trecho percorrido pelo ônibus em cada logradouro da rota. Primeiramente, são identificados os cruzamentos onde o ônibus faz a conversão do logradouro anterior para o seguinte. Para um itinerário $I = \{l_1, l_2, ..., l_n\}$ (uma lista de logradouros ordenados segundo a ordem em que são visitados no trajeto), o Algoritmo 2 tenta encontrar o cruzamento entre cada par de logradouros l_i e l_{i+1}. Um *cruzamento* é definido como dois arcos, a e b, tal que a está associada ao logradouro l_i, b está associada ao logradouro l_{i+1} e a e b se interceptam (são arcos adjacentes na malha viária).

Entrada: Tabela Trecho (Alg. 1), Itinerário I = $(l_1, l_2, ..., l_n)$
Saída: Tabela C de arcos de cruzamento

1 **para todo** $(l_i, l_{i+1}) \in I$ **faça**
2 | $C(l_i, l_{i+1}) \leftarrow \{(a, b) : a \in l_i, b \in l_{i+1}, intercepta(a, b)\}$;
3 **fim**

Algoritmo 2: *Identificação dos cruzamentos em I*

Dado o logradouro l_i de um itinerário I, se os cruzamentos (l_{i-1}, l_i) e (l_i, l_{i+1}) forem encontrados, é possível identificar o trecho do logradouro l_i que é percorrido pelo ônibus através de uma busca em largura sobre a malha viária. Apresentado no Algoritmo 3, o procedimento difere de uma busca em largura tradicional por existirem potencialmente

várias fontes (todos os arcos do primeiro cruzamento) e destinos (os arcos do segundo cruzamento) e pela busca ser restrita a arcos associados a l_i. No presente trabalho, qualquer caminho (trajeto de uma aresta $f \in fonte$ até outra aresta $d \in destino$) encontrado entre os dois conjuntos é aceito, pois fornece uma geometria bastante razoável do segmento de logradouro percorrido pelo ônibus. A Figura 4 mostra um exemplo de aplicação do Algoritmo 3.

Entrada: Tabela Trecho (Alg. 1), Tabela C (Alg. 2), logradouro l_i
Saída: Conjunto de arcos percorridos em l_i

$fonte \leftarrow \{b : (a, b) \in C(l_{i-1}, l_i)\}$;
$destino \leftarrow \{a : (a, b) \in C(l_i, l_{i+1})\}$;
$fila \leftarrow fonte$;
enquanto *há arcos não visitados na* $fila$ **faça**
    $u \leftarrow$ próximo elemento da $fila$;
    se $u \in destino$ **então**
        Retorna o caminho entre $fonte$ e u;
    fim
    $fila \leftarrow fila \cup \{v : intercepta(u, v), v \in l_i, v \notin fila\}$;
fim
Retorna $\varnothing$;

Algoritmo 3: *Reconstrução do trecho percorrido no logradouro* l_i

Figura 4. Cruzamentos entre o Anel Rodoviário e a Avenida Antônio Carlos (vermelho) e as Avenidas Antônio Carlos e Bernardo Vasconcelos (verde). No Algoritmo 3, os arcos vermelhos são a fonte, os verdes o destino e o algoritmo é capaz de encontrar um caminho qualquer conectando fonte e destino através de uma busca em largura restrita aos arcos da Avenida Antônio Carlos.

5. Aplicação online

Utilizando o método apresentado na Seção 4, foram geradas geometrias para os itinerários de todas as linhas de ônibus coletadas. Os itinerários foram exportados para arquivos KML e uma visualização construída utilizando a API do Google Maps[5].

Os trajetos apresentados podem ter trechos de duas cores. Trechos em verde são aqueles recuperados segundo o método proposto, ou seja, trechos de logradouros entre dois cruzamentos que puderam ser identificados com base no itinerário oficial da BHTRANS e no processamento da malha viária. Trechos em laranja indicam logradouros presentes no itinerário e encontrados no banco de dados, mas para os quais não foi possível identificar um trecho específico entre dois cruzamentos; neste caso, o logradouro inteiro é destacado. Em trabalhos futuros, serão avaliadas heurísticas para recuperação da geometria dos trechos em laranja, tais como a seleção de arcos próximos a cruzamentos identificados, incorporação dos pontos de ônibus ou busca por caminho mínimo.

[5]O sistema desenvolvido, que permite ao usuário selecionar uma linha e itinerário de ônibus e visualizar o trajeto correspondente no mapa, pode ser acessado no endereço `http://aqui.io/onibusBH`.

6. Conclusão

A partir da disponibilidade de uma representação geográfica de cada linha de ônibus, passa a ser possível planejar o desenvolvimento de novas ferramentas. Por exemplo, é possível indicar as linhas de ônibus que atendem a um determinado local, ou mesmo implementar algoritmos de otimização em redes para obter rotas ideais para o deslocamento usando apenas transporte coletivo [Zuppo et al. 1996]. O planejamento detalhado de deslocamentos é parte importante de sistemas de apoio à mobilidade urbana, frequentemente incluídos no repertório de Cidades Inteligentes. Diante da complexidade e da frequência de mudanças no transporte coletivo, um sistema adequado de apoio ao cidadão não pode ser construído sem acesso livre aos dados mais atuais, gerenciados pelas autoridades competentes.

A motivação para o desenvolvimento do presente trabalho veio em parte da constatação de que os dados do sistema de transporte coletivo, embora existentes e gerenciados em meio digital, não são publicados amplamente e em formato legível por máquina, como seria de se desejar desde a publicação da Lei de Acesso à Informação (Lei 12.527/2011). Caso os dados estivessem disponíveis, por exemplo, através de serviços Web baseados na versão mais atual do sistema de transporte coletivo, academia e empresas poderiam propor e implementar soluções inovadoras de apoio ao cidadão em sua necessidade de mobilidade nas cidades [Work and Bayen 2008]. Portanto, consideramos que o ideal seria que o presente trabalho fosse tornado desnecessário o quanto antes, pela publicação dos dados, continuada e com qualidade, em um meio adequado e formato tecnologicamente neutro.

O trabalho aqui apresentado pode ser continuado de diversas formas, como por exemplo (1) pela adaptação a outras cidades, (2) pela expansão em direção ao uso de técnicas de otimização para planejamento completo de rotas, e (3) pela adaptação a plataformas móveis, como smartphones e tablets. Pode-se também conceber a integração com sistemas de coleta de dados fornecidos voluntariamente (*volunteered geographic information*, VGI) de modo que os usuários possam indicar erros nos dados e trocar impressões sobre a qualidade do serviço de transporte público.

Agradecimentos

Os autores agredecem ao CNPq (308678/2012-5) e FAPEMIG (CEX-PPM-00518/13) pelo apoio no desenvolvimento deste projeto, e à Empresa de Informática e Informação do Município de Belo Horizonte (PRODABEL) pela cessão dos dados de logradouros utilizados no trabalho.

Referências

Cardoso, L. (2007). Transporte Público, Acessibilidade Urbana e Desigualdades Socioespaciais na Região Metropolitana de Belo Horizonte. Master's thesis, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brazil.

Work, D. B. and Bayen, A. M. (2008). Impacts of the Mobile Internet on Transportation Cyberphysical Systems: Traffic Monitoring using Smartphones. In *National Workshop for Research on High-Confidence Transportation Cyber-Physical Systems: Automotive, Aviation and Rail*, Washington, DC, USA.

Zuppo, C., Davis, C., and Meirelles, A. (1996). Geoprocessamento no Sistema de Transporte e Trânsito de Belo Horizonte. In *GIS Brasil 96*, pages 376–387.