Processing Collections of Geo-Referenced Images for Natural Disasters

Fernando Loor[1,2], Veronica Gil-Costa[2] and Mauricio Marin[1]

[1] *Universidad Nacional de San Luis, Argentina.*
{fernandoloor1, gvcosta}@unsl.edu.ar
[2] *Universidad de Santiago de Chile, Chile.*
mauricio.marin@usach.cl

Abstract

After disaster strikes, emergency response teams need to work fast. In this context, crowdsourcing has emerged as a powerful mechanism where volunteers can help to process different tasks such as processing complex images using labeling and classification techniques. In this work we propose to address the problem of how to efficiently process large volumes of georeferenced images using crowdsourcing in the context of high risk such as natural disasters. Research on citizen science and crowdsourcing indicates that volunteers should be able to contribute in a useful way with a limited time to a project, supported by the results of usability studies. We present the design of a platform for real-time processing of georeferenced images. In particular, we focus on the interaction between the crowdsourcing and the volunteers connected to a P2P network.

Keywords: Geo-referenced images, support platforms for natural disaster, P2P network.

1. Introduction

During the last years, Big Data has become a strong focus of global interest, attracting more and more attention from academia, industry, government and other organizations. The growing flow of data, which comes from different types of sensors, messaging systems and social networks, in addition to more traditional measurement and observation systems, has already invaded many aspects of our daily existence.

Large data, including large geo-referenced data, have great potential to benefit many social applications such as climate change, health, surveillance, disaster response, critical infrastructure monitoring, transport, etc. Geo-referenced data describe elements in relation to the geographical space - location, often with location coordinates in a spatial reference system. The term commonly used for systems that use this type of data is known as GIS - Geographic Information System. The geo-tagged contents of the web, voluntary geographic information (VGI), satellite navigation, etc. are traditionally collected through sensors. [5]. The authors in [9] claim that geo-referenced data from social networks is another form of VGI data.

The information that is posted on social networks can be very useful in case of a natural disaster. As direct witnesses of the situation, people share photos, messages and videos about events that get their attention. In an emergency operations center, these data can be collected and integrated into the management process to improve the general understanding of the situation or rescue actions. This type of volunteer participation is known as crowdsourcing. Digital volunteers have helped to collect pertinent information much faster than officials or people in charge of coordination activities in natural disasters could do alone, with huge potential impacts on the responsibilities of officials in the management of the information. Emergency search and rescue teams already use pre-event remote sensing data when planning operations [16].

In general, the increasing volume and variable format of the large georeferenced data collected pose additional challenges in storage, management, processing, analysis, visualization and verification of data quality. The authors in [14] state that the size, variety and update rate of data sets exceed the capacity of spatial computing technologies and spatial databases commonly used to learn, manage and process data. with a reasonable effort.

This paper describes the design of a distributed platform that combines algorithms, processing techniques and associated software tools to efficiently

label large volumes of georeferenced images from different sources (satellite images, captured by unmanned aerial vehicles, originated in social networks), to optimize the work of digital volunteers in situations of natural disasters in order to build status maps in real time. In particular, we focus on the interaction between the crowdsourcing server receiving new images to be analyzed and the users inter-connected in a P2P network. The crowdsourcing server sends tasks to the peer-volunteers which tags the images or just vote for an option from a given option list. The crowdsourcing server gathers all the votes form the peers and evaluate their contributions. If there is an agreement in the voting (e.g. most of the volunteers vote for the same option), the task is finished, and the results are saved in a database for statistic purposes. Otherwise, the server selects volunteers with a high reputation and sends the task again.

The remaining of this paper is organized as follows. Section 2 presents previous works. In Section 3 we present the design of our proposed platform and the simulation model for the interaction between the crowdsourcing server and the volunteers. Section 4 presents experimental results and Section 5 concludes.

2. Previous works

New forms of georeferenced data collection have emerged that have given rise to completely new data sources and data types of a geographical nature. The data acquired by the public -VGI-, and the data from the geo-sensor networks have led to a greater availability of spatial information. Whereas until recently, the authoritarian data sets dominated the topographic domain, these new types of data expand and enrich geographic data in terms of thematic variation and the fact that they are more user-centered. The latter is especially true for VGI compiled by social media [13].

Some authors claim that "80% of the data is geographical in nature" [8]. Much of the data in the world can be georeferenced, which indicates the importance of georeferenced large data management. The georeferenced data describe objects and things in relation to the geographical space - location - often with coordinates of location in a spatial reference system. The term commonly used for systems that use this type of data is known as GIS - Geographic Information System.

The work in [3] presents a study of how to find out the current locations of users by tracking their mobile devices, such as smartphones. The study mentions services of geo-social networks such as Foursquare used to locate friends and to find nearby shops and restaurants, where many users register in several places and reveal their current location. In other words, it proposes to solve queries using information from different individuals through its mobile devices.

The work presented in [2], describes some examples of how the Department of Defense of the United States uses crowdsourcing to give answers to problems of natural disasters. The authors conclude that there is a great benefit in taking advantage of the power of the crowd, "a process that will continue to mature, evolve and define the way we help others today, tomorrow and in the future."

Barrington et al. [1] presents a review of the state of the art on the use of crowdsourcing and analysis of images, particularly high resolution aerial. This work describes the experiences obtained in the cases of the earthquake in Haiti and in 2008 in China. (Lee and Kang, 2015) describe the impact of georeferenced data in different applications such as marketing and propose a three layers platform to index and analyze images. But this proposal does not consider the collaboration of volunteers.

Ofli et al. [10] propose a hybrid scheme based on automatic techniques and crowdsourcing for aerial image processing. In this case, manual annotations are used to train a supervised learning system. However, this work does not describe the platform used and does not consider the interaction with information collected by people who are in the place of the event.

There are some platforms such as Tomnod (http://www.tomnod.com), GeoTag-X [15], and some research works that address the problem of using volunteers for the processing of georeferenced images [1] [18] [17] [11] [4]. In particular, Tomnod of the company DigitalGlobe is using Artificial Intelligence (AI) driven by crowdsourcing to automatically identify the characteristics of interest in satellite and aerial images. Tomnod runs crowdsourcing campaigns, where volunteers support data mapping by validating the results of an image mining algorithm. GeoTag-X is a research project aimed at researching and evaluating collaborative on-line environments and software tools for creative learning.

3. Proposed methodology

In this section, we present our platform design for processing large number of georeferenced images. Our design focused on real-time applications devised for natural disasters which require a low latency and short response time.

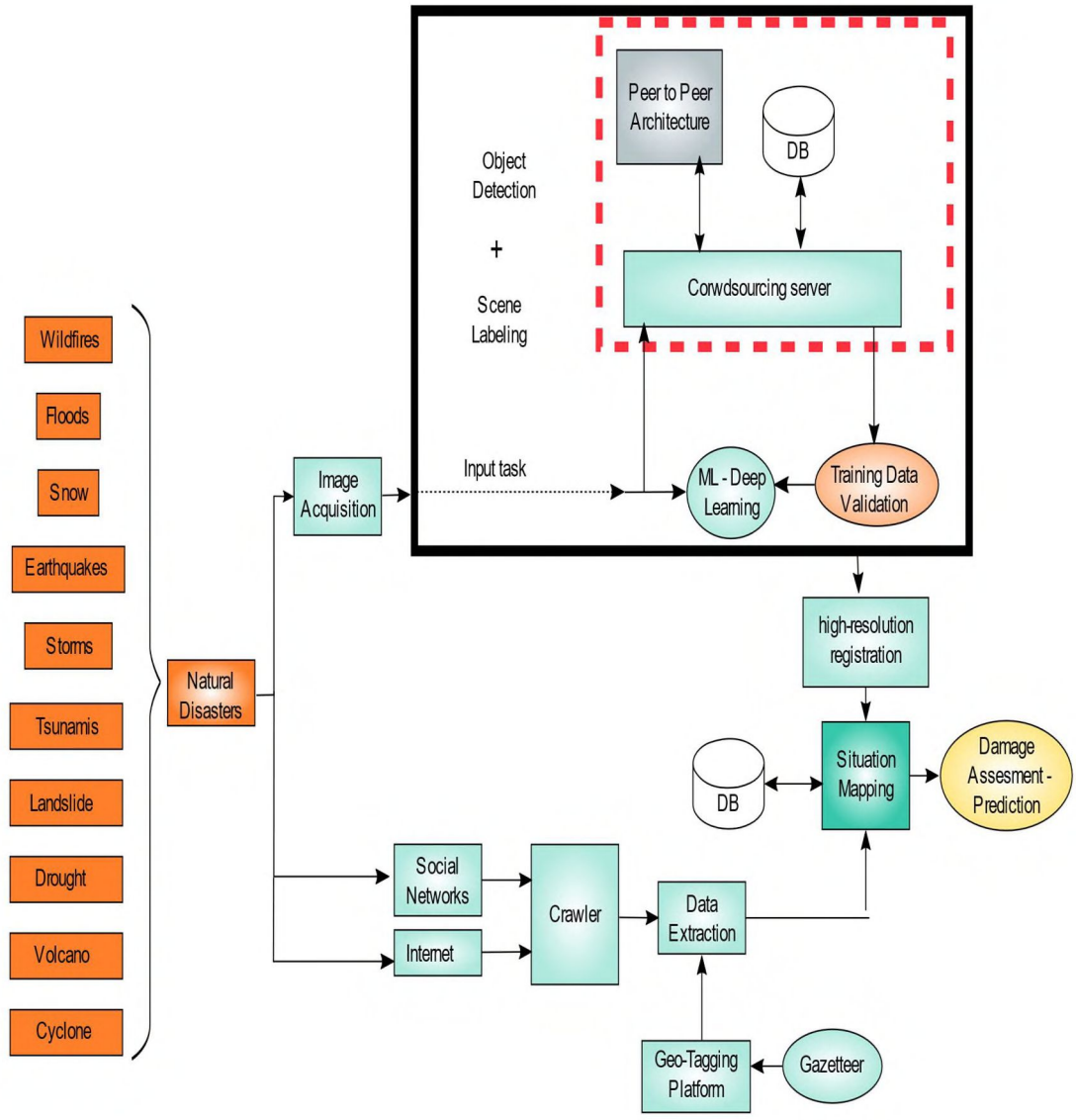


Fig. 1: Proposed distributed platform.

Fig. 1 shows the general components of our proposed platform and how they communicate to each other. Upon a natural disaster, images are captured through different devices such as a drone and smartphones (*Image acquisition*). Then, these images are sent to the *Object detection and scene labeling* component – the black box at the top of the figure. For each incoming image we create a task which will be sent to the crowdsourcing server or to the Machine Learning server. The task contains metadata associated with the image like the GPS coordinates, the image identifier, and a list of options to vote or to label the image. If the machine learning server is not trained for the region where the images come from, the task is sent to the crowdsourcing server. This server will request to different volunteers to help tagging the images. It allows different users to classify images manually, in order to generate new classifiers in real time. Classifiers can detect emerging needs at the time of a disaster.

The results obtained from the crowdsourcing server are used to train the machine learning server. Once this last server is trained, it can start receiving incoming tasks. In this way, we can reduce the amount of work sent to volunteers.

The results achieved by the object detection and scene labeling component are sent to the high-resolution registration component. This last, establishes a correspondence between the image been analyzed and the geographical coordinates to create a situation map. The information inside the situation map is stored in a NoSQL database like MongoDB and can be later use for emergency management and

help for victims.

At the bottom of the figure, we show the components deployed in our proposed platform for text messages crawled from Tweeter or others social networks. Relevant data is extracted from these texts The Geo-Tagging Platform uses the Gazetteer which is a geographic dictionary to parse the tweets to identify which coordinates they belong to.

3.1 Modelling the crowdsourcing server

Fig.2 shows the general scheme of our model for the crowdsourcing server and the P2P network. Our model considers a group of users who execute the tasks delivered by the crowdsourcing server. Each task has information related to the incoming image like the time to live (TTL) of the task, the priority, the image identifier, and the list of options. If the priority feature is on, the priority task is high and should be process immediately.

Peers build an overlay network managed by the Internet service providers (ISP). In particular, our model follows a P2P Distributed Hash Table - DHT [12].

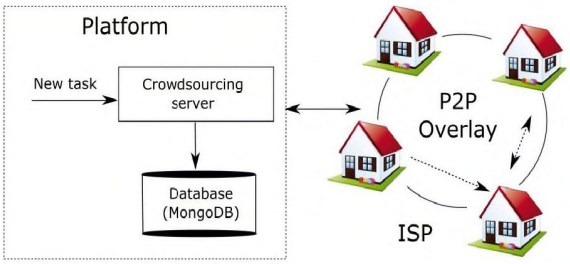


Fig. 2 Simulated scheme.

Internet service providers (ISPs) are responsible for delivering Internet access to clients from a given geographic area. To communicate with other ISPs, it is necessary to access the Internet backbone. The backbone is a shared network which enables communication among ISPs of the world. To make use of the network, ISPs must respect a Service Level Agreement (SLA) contract in which they commit to regulate their traffic to not compromise another ISPs communication.

Each peer holds a task queue which stores tasks with the priority feature given by the TTL. These tasks should be quickly process in order of arrival.

In general, the sequence of steps executed by our model is as follows:

(1) The crowdsourcing server receives incoming tasks from the platform.
(2) A peer becomes visible to the server indicating that he wants to be a volunteer.
(3) The server sends a list of tasks to the peer.
(4) The peer agrees to start processing the tasks and request the first image.
(5) The server sends the first image to the peer and set a time-to-live (TTL).
(6) The peer executes the task and sends the result to the server.
(7) If a total of **H** answers were received for a task, the server evaluates if there is an agreement in the voting.
    a. If so, the task is finished, and the data is sent to a data base server like MongoDB or Casandra.
    b. Otherwise, the server selects new peers with high reputation to send the task again.
(8) If the peer wants to continue collaborating, return to step 4 for the next images.

To reduce the communication between the crowdsourcing server and the P2P network, each peer has an LRU cache memory with images received from the server. Thus, the next time a peer requests an image form the same task list, the peer will search for that image inside the P2P network using the DHT before sending a request to the server.

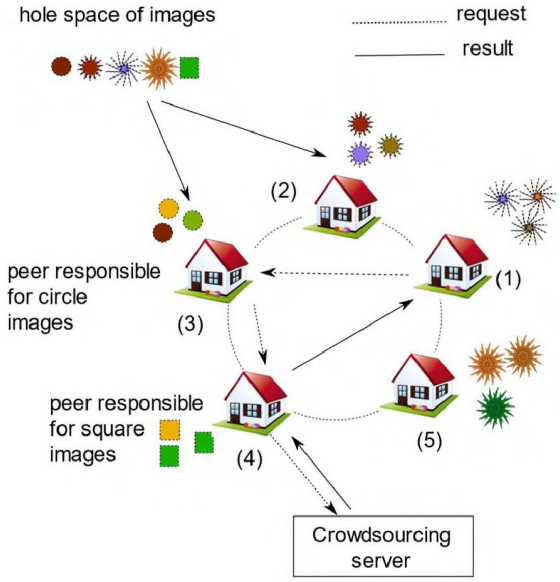


Fig. 3 Communication scheme.

More precisely, as the hole space of images is divided among the peers, each peer is responsible for a particular set of images. As shown in Fig. 3, each peer is responsible for a different set of images (square, circle and the three types of stars). In this example, the peer 1 request a square image of color blue. Then it sends a message to peer 3 which is selected by the DHT. Due to peer 3 does not have the requested image, it sends the message to the next hop in the P2P network. In this case, the message reaches to peer 4, which is the peer responsible for this kind

of images. As peer 4 does not have the blue square image, it sends a request to the crowdsourcing server. The server sends the requested image to the peer 4, which is the responsible. Peer 4 insert this new image into its LRU cache memory. Finally, peer 4 sends the image to the requesting peer 1.

If the TTL of a given task expires, the server sends the results achieved at the moment to the database server. Additionally, some volunteers can offer to continue helping with the voting/tagging tasks. In this case, the value of **H** – number of expected answers per task- can be increased. This helps to improve the quality of the results and get a more adequate consensus of the tasks of voting/tagging.

The server applies two policies for the TTL of each task. (1) If the TTL of a given task expires and there are at least **H** results received from the peers, the server ranks the results. (2) Otherwise, if the TTL expires and there are less than **H** results, the server extends the TTL and sends additional request with high priority (the priority featured turned-on) to frequently active volunteers.

4. Experimental results

In this work, we evaluate the performance of the crowdsourcing server and its interaction with the network of volunteer forming a P2P network. We have built a simulator that implements a transport layer, a P2P overlay and our caching proposal. Pastry [12] has been used as the overlay network in our experimentation. We have also simulated the Web crowdsourcing server, a generator which creates the incoming tasks and a database server which receives data form the crowdsourcing server. The simulation is divided in simulation time windows of 100 units of time.

4.1. Simulation approach

The simulation model of this work uses a processes and resources approach. Processes represent the crowdsourcing server, and the peers in charge of transaction processing. Resources are artifacts such as the data of the incoming messages, global variables like the input queue of each process and also the CPU and the communication network. The simulation program is implemented using LibCppSim [7], where each process is implemented by a co-routine that can be blocked and unblocked at will during simulation.

The operations *hold*(), *passivate*() and *activate*() are used for this purpose. Thus, a coroutine C_i can be paused for a given amount of time Δ_t -which represents the duration a task. Once the simulation time Δ_t has expired, the coroutine C_i activates itself if a hold() operation was previously executed. Otherwise, the coroutine C_i is activated by another coroutine C_j using the activate() operation. This last case allows to represent the interaction among the different components of the simulated platform.

The simulated architecture assumes a classical DHT overlay composed by **N** physical nodes (or peers) and **K** object-keys (task with images ids) mapped onto a ring. Objects (images) stored in a DHT such as Pastry [12], have a responsible peer in the network that is the peer with the closest ID to the key of the object. Thus, any given peer is responsible for a fraction of these images and (on request) it must contact the crowdsourcing server to get them.

4.2. Experiment Settings

We evaluate different metrics and costs between the crowdsourcing server and the peers by setting the parameters of our simulator as shown in Table 1.

Table 1 Parameters setting for the simulation.

Parameter	***Value***
Network size (num. peers)	{100,500,1000, 1500}
Cache size in each peer	{50,100,150}
Total answers expected by the crowdsourcing server (**H**)	{10,15,20,25}
Arrival rate	{500, 1000, 2000}

4.3. Gini Coefficient

To measure load balance among peers we use a metric based on Lorenz curves called the Gini coefficient, which is a metric commonly used on other fields like economics and ecology.

If all peers have the same load, the Lorenz curve is a straight diagonal line, called the line of equality or the perfect load balancing. If there is any imbalance, then the Lorenz curve falls below the line of uniformity. The total amount of load imbalance can be summarized by the Gini coefficient G, which is defined as the relative mean difference, i.e. the mean of the difference between every possible pair of peers, divided by their mean load. G values range from 1 to 0. The value 0 is achieved when all peers have the same load. The value 1 is achieved when one peer receives the whole system load while the remaining peers receive none. Therefore, when G approaches 0, global load imbalance is small, and when G approaches 1 the imbalance is large.

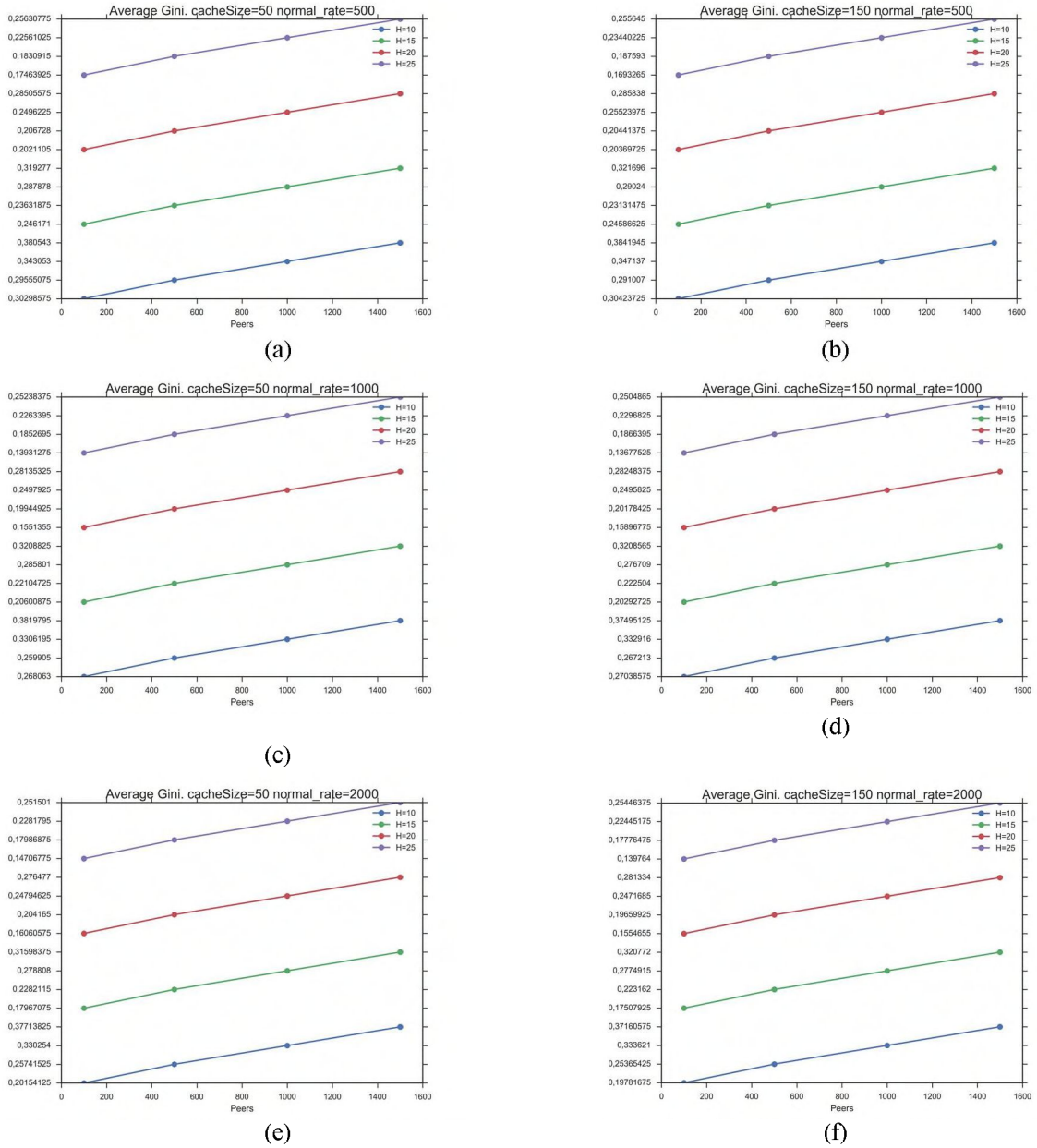


Fig. 4: Gini coefficient for different cache size, number of peers, arrival rate and different values of H.

In Fig. 4, the y-axis shows the Gini coefficient for different configurations. The x-axis shows the values of H. In Fig. 4 (a)(b)(c)(d)(e) and Fig. 4(f) we vary the arrival rate from 500 to 2000 requests per seconds and the cache size from 50 to 150. Results obtained with a cache size of 100 are very similar to the ones presented in these figures.

In all cases, the Gini coefficient is lower than 50%, meaning that the workload tends to be balance among the peers. Moreover, as we increase the value of H (in the x-axis) the imbalance tends to be lower, as more peers execute similar amounts of tasks.

On the other hand, the arrival rate and the cache size have very low impact on the performance achieved by the peers. However, as we increase the number of peers, the workload tends to be more balanced. That's because, the tasks are evenly distributed among the peers and the query arrival rate is high tending to produce different queue size inside each peer.

4.4. Number of Hops

Fig. 5 shows the average number of hops that a massage has to go thought, inside the P2P network, before reaching the requested image. The x-axis shows the number of peers selected by the crowdsourcing server to send the tasks. We show the results obtained for a cache size of 150 and an arrival rate of 2000 tasks per second. Results obtained with other parameters are very similar, because the number

of hops does not depend on the cache size of the peers, neither on the arrival rate.

As expected, the number of hops tends to increase with a larger number of peers in the network. On the other hand, the H value impacts on the number of hops. A larger H value tends to reduce the number of hops required to find the image inside the P2P network. That's because, as more peers are involved in the process to solve a task (e.g. voting or tagging), the images for those tasks are going to be used by more peers, and therefore a new peer requesting an image would probably find that image in a near by peer.

With few peers, the H value does not impact on the number of hops. That's because the network size is small and the peer for a give image is found more quickly, without having to visit other peers along the way.

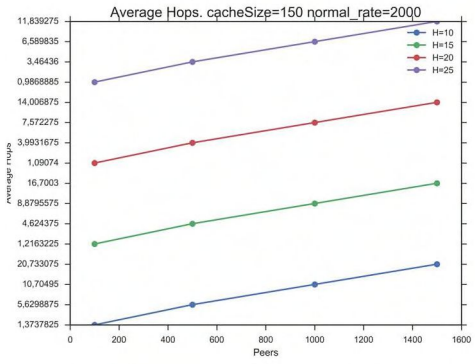


Fig. 5: Average number of hops acieved by different parameters of our simulated platform.

4.5. Latency

In Fig. 6, we show the average latency reported by sending messages inside the P2P network. We show results for different arrival rates and a cache size of 150. Notice that the cache size does not impact on the latency.

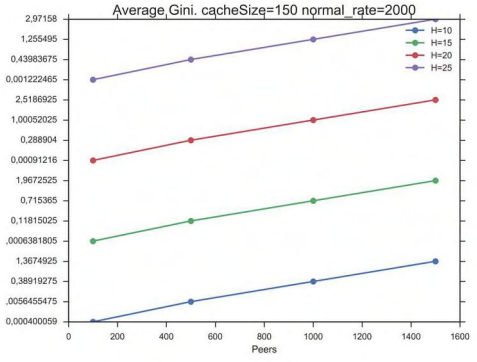
As expected, more peers in the network tends to increase the latency, due to more hops are visited before reaching the final destination. That is, the peer holding the requested image.

Also, the arrival rate tends to smoothly increase the latency in the P2P network, due to more messages are present in the network at the same time and those messages compete for the network resource.

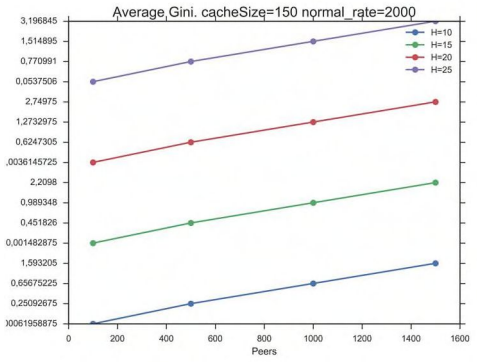
Finally, with a larger H value, the latency also increases because more peers have to request the same tasks and images to process. These requests are messages traversing the P2P network.

5. Conclusions

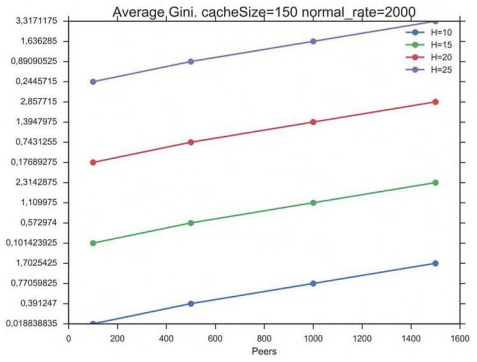
In this work, we presented a new platform to support decision making in cases of natural disasters, through the processing of information such as georeferenced images in real time. We presented the components of our proposal and how they communicate to each other.



(a)



(b)



(c)

Fig. 6: Latency reported by the P2P network for different arrival rates (a) 500, (b) 1000 and (c)

We also modelled the interaction between the crowdsourcing server and the volunteers forming a P2P network. We simulated this model to analyze possible bottlenecks and the benefits of using cache memories in the peers to avoid accessing the crowdsourcing server for each new requested task.

We evaluated different metrics to analyze the effect of communication and the number of peers volunteers on the performance of the platform.

Results show that the number of selected peers as volunteers affects the latency of the communication in the P2P network as the number of peers increases.

Acknowledgements

This research was partially funded by PICT 2014 N° 2014-01146.

Competing interests

The authors have declared that no competing interests exist.

References

[1] L. Barrington, G. Shubharoop G. Marjorie, H.N. Shay, J. Berger, S. Gill, A. Yu-Min Lin and C. Huyck. "Crowdsourcing earthquake damage assessment using remote sensing imagery". Annals of Geophysics. vol. 54, no. 6. ,2011

[2] D. Becker and S. Bendett. "Crowdsourcing Solutions For Disaster Response: Examples And Lessons For The US Government, Humanitarian Technology: Science, Systems and Global Impact", in HumTech, 2015.

[3] M. Choy, J.-G. Lee, G. Gweon, D. Kim. "Glaucus: exploiting the wisdom of crowds for location-based queries in mobile environments", in Proceedings of 8th In-ternational AAAI Conference on Weblogs and Social Media, pp.61–70, 2014.

[4] P. Díaz, J. M. Carroll and I. Aedo. "Coproduction as an Approach to Technology-Mediated Citizen Participation in Emergency Management", in Journal of Future Internet, vol. 3, no. 3, 2016.

[5] M. R. Evans, D. Oliver, X. Zhou and S. Shekhar. "Spatial Big Data: Case Studies on Volume, Velocity, and Variety". in H. A. Karimi (Ed.), Big Data: Techniques and Technologies in Geoinformatics, pp. 149-176, 2014

[6] L. Jae-Gil and K. Minseo. "Geospatial Big Data: Challenges and Opportunities", in Big Data Research, pp. 74–81, 2015.

[7] M. Marzolla. "Libcppsim: a Simula-like, portable process-oriented simulation library in C++", in ESM, pp. 222–227, 2004.

[8] C. D. Morais. "Where is the Phrase "80% of Data is Geographic" From?". from http://www.gislounge.com/80-percent-data-is-geographic/, 2012.

[9] S. Newsam. "Crowdsourcing What Is Where: Community-Contributed Photos as Volunteered Geographic Information", in Knowledge Discovery from Community- Contributed Multimedia, pp 36-45, 2010.

[10] F. Ofli, P. Meier, M. Imran, C. Castillo, D. Tuia, N. Rey, J. Briant, P. Millet, F. Reinhard, M. Parkan, and S. Joost. "Combining human Computing and Machine Learning to Make Sense of Big (Aerial) Data for Disaster Response", in Journal of Big Data, vol. 00, no. 00, pp 1-13, 2016.

[11] T. Onorati, P. Díaz. "Giving meaning to tweets in emergency situations: a semantic approach for filtering and visualizing social data", in Journal of SpringerPlus, vol. 5, no. 1, 2016.

[12] A. Rowstron and P. Druschel, "Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems," in Middleware, ser. LNCS, vol. 2218, pp. 329–350, 2001.

[13] M. Sester, J.J., Arsanjani, R.., Klammer, D. Burghardt and J-H. Haunert. "Integrating and Generalising Volunteered Geographic Information Abstracting Geographic Information in a Data Rich World" in *Methodologies and Applications of Map Generalisation*, pp. 119-155, 2014.

[14] S. Shekhar, V., Gunturi, M.R., Evans, and K. Yang. "Spatial Big-Data Challenges Intersecting Mobility and Cloud Computing", in Proceedings of the 11th ACM International Workshop on Data Engineering for Wireless and Mobile Access – MobiDE, 2012.

[15] C. Smith. "A Case Study of Crowdsourcing Imagery Coding in Natural Disasters", in Data Analytics in Digital Humanities, pp. 217-230, 2017.

[16] S. Thorvaldsdóttir, E. Birgisson and R. Sigbjornsson. "Interactive on-site and remote damage assessment for urban search and rescue", Journal of Earthq. Spectra, vol. 27 no. (S1), pp S239-S250, 2011.

[17] C. Turk. "Cartographica incognita: 'Dijital Jedis', Satellite Salvation and the Mysteries of the 'Missing Maps'", Journal of The Cartographic, pp. 14-23, 2016.

[18] N. Witjes, P. Olbrich and I. Rebasso. "Big Data from Outer Space: Opportunities and Challenges for Crisis Response", in Yearbook on Space Policy 2015: Access to Space and the Evolution of Space Activities, pp 215-225, 2017.