Draft version. To appear in Int. Conf. on Informatics in Control, Automatic and Robotics, ICINCO 2013

Building and Exploiting Maps in a Telepresence Robotic Application

Javier Gonzalez-Jimenez, Cipriano Galindo, Francisco Melendez-Fernandez, and J.R. Ruiz-Sarmiento
System Enginnering and Automation Dpt., Malaga, Spain
{javiergonzalez,cgalindo,fco.melendez,jotaraul}@uma.es

Keywords: Telepresence Robotics, Mapping, Assistive Robotics, Teleoperation.

Abstract: Robotic telepresence is a promising tool for enhancing remote communications in a variety of applications. It enables a person to embody a robot and interact within a remote place in a direct and natural way. A particular scenario where robotic telepresence demonstrates its advantages is in elder telecare applications in which a caregiver regularly connects to the robots deployed at the apartments of the patients to check their health. Normally, in these cases, the caregiver may encounter additional problems in guiding the robot because s/he is not familiar with the houses. In this paper we describe a procedure to remotely create and to exploit different types of maps for facilitating the guidance of a telepresence robot. Our work has been implemented and successfully tested on the Giraff telepresence robot.

1 INTRODUCTION

In the last years, *robotic telepresence* is receiving a great deal of attention from the robotic community, especially when applied to the social interaction of the elderly (Coradeschi et al., 2011; Tsui et al., 2012). Robotic telepresence refers to a combination of technologies that enables a person to be virtually present and to interact in a remote place by means of a robot. Briefly, a *visitor* takes the control of a mobile robot that physically interacts with the *user* that receives the service (see figure 1). The result is that the user identifies somehow the robot as the person who is controlling it, i.e. the visitor, and establishes a social relation as s/he was actually in the place. A typical scenario where robotic telepresence becomes relevant is its utilization by healthcare personnel, e.g. nurses and doctors, to carry out professional visits to a number of patients to check their general health and mental state from anywhere. In these cases, as well as in other situations where the visitor is not familiar with the house, it is of a great help to provide the visitor with a schematic map of it where the real-time position of the robot is displayed.

Considering maps of the environment in robotic teleoperation is a generally neglected issue: it is assumed that the human abilities are enough for guiding a robot even if the environment is unknown. However, the advantages of enhancing the graphical teleoperation interface with a map are clear in terms of safety, convenience and efficiency of the robot teleoperation.

In this work we present an intuitive and interactive process that permits the visitor, i.e. the person who drives the robot, to create and productively exploit maps in a telepresence application.

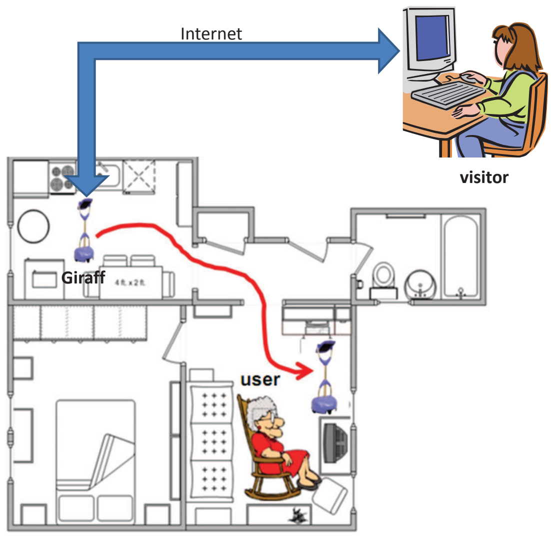


Figure 1: Robotic telepresence application. The visitor remotely drives the robot deployed in the user's apartment and interacts with her through videoconferencing.

The developed work is framed in the project ExCITE –Enabling SoCial Interaction Through Embodiment– (Coradeschi et al., 2011) under the Ambient Assisted Living European Joint Programme and GiraffPlus –Combining social interaction and long term monitoring for promoting independent living– funded by EU within the FP7th. Within such projects, several prototypes of a telepresence robot called *Giraff* (see figure 2) have been deployed at the elders' homes, enabling healthcare personnel and relatives to interact with them. Initial results from the evaluations on the use of the Giraff robots by non-technological users reveal that in spite of the clear benefits of telepresence robots, there are still some hurdles that complicate the commercial deployment of this technology. A significant and recurrent limitation reported by the visitors, is the disorientation they suffered when they teleoperate the robot, especially in large or unknown environments. This problem worsens when the visitor is a caregiver that visits a number of patients.

This paper addresses this issue and proposes an intuitive map building mechanism that permits a non-tecnological visitor to construct a geometric-topological map of the environment while teleoperating the robot. The obtained map is used for two purposes. First, to localize the robot in real-time by applying well-known robotics techniques, and second, extract from it a schematic plan which is integrated into the graphical interface to display the pose of the robot within the apartment. This map also enables the visitor to give high-level navigational commands to the robot, e.g. "go to the kitchen", if the robot is featured with autonomous navigation algorithms. The approach presented here extends a previous work (González-Jiménez et al., 2012) that addressed a number of improvements on the Giraff telepresence robot, including a preliminary solution for mapping and localization. The major differences and new contributions of the presented work w.r.t. the previous ones are:

- An interactive method for map building specially targeted to non-technological users.
- The map building process is completely carried out at the visitors' side.
- The visitor can easily update the entire map or parts of it when needed.

The structure of the paper is as follows. Section 2 describes the Giraff telepresence robot. Section 3 gives a general overview of the proposed map building process. Next, section 4 presents the software architecture and modules developed in our implementation. Finally some conclusions and discussions on the advantages of exploiting maps in robotic telepresence are outlined.

2 The Giraff Telepresence Robot

The Giraff robot, or simply Giraff, is a telepresence robot developed by the Giraff AB company (Giraff, 2013). It consists of a motorized wheeled platform endowed with a videoconferencing set, including camera, microphone, speaker and screen. Giraff permits a virtual visitor to move around, perceive the environment (audio and video), and chat with the user. The height of Giraff, the streaming of the visitor camera on the screen, and the possibility of tilting the Giraff's head help in establishing a friendly interaction with the user who can experience that the visitor is at home.

Figure 2: The Giraff telepresence robot equipped with a laser range scanner for map building and localization.

From a technical point of view, Giraff relies on a low-cost, commercial computer onboard. The batteries of Giraff last, approximately, two hours and are charged by docking the robot at a station plugged to a normal wall socket of the house.

The Giraff manufacturer provides a software application, called the Giraff *Pilot*, to easily teleoperate the system. Pilot, is essentially a graphical interface for driving the robot and controlling the standard videoconference options, i.e., to initiate/hang-up a call, and to adjust the speaker and microphone volume (see figure 3). At the Giraff side, a server is continually running, accepting calls and providing the needed functionality for videoconferencing and motion commands. All the actions needed from the elder to handle Giraff can be very easily accomplished with a remote controller. Thus, one of the major advantages of the Giraff telepresence robot is that neither the user nor the visitor need any technological skill to use it, and they both can manage the system (Pilot and Giraff) in an intuitive and natural way.

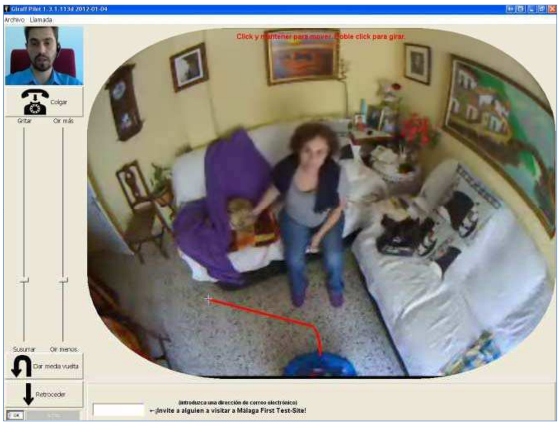


Figure 3: The teleoperation interface Pilot. The visitor guides the robot by drawing the desired trajectory on the screen.

In order to feature the commercial version of the Giraff robot with the capability of building a map of the house and compute its position in it, the robot has been equipped with a proper 2D range laser scanner URG-04LX-UG01 (Hokuyo, 2013) attached on its as shown in figure 2. This type of sensors have a mature technology, widely used in robotics systems for carrying out mapping, localization and obstacle detection tasks. The main characteristics of the selected model are: a field of view of 240 degrees with a resolution of 0.36, an operational range up to 4 metres and a working frequency of 10 Hz.

3 The Map Building Process

The map building process presented in this work involves de following steps:

1. The visitor initiates the mapping process through the corresponding button in the client interface (see figure 4b), being then requested to drive the Giraff robot within the house, visiting all the rooms to be included in the map. During the navigation, the robot odometry and the readings from the scan laser are continuously gathered and sent to the client using the MQTT protocol (Hunkeler et al., 2008).
2. When the visitor decides to finalize the map construction (switching off the “build map” button), an implementation of the ICP algorithm (Besl and McKay, 1992) is run in his computer to register all the received scans, generating a point-based map. This *geometric map* is sent to the robot, which will use it for localization purposes (Blanco et al., 2010).
3. The resultant geometric map is presented to the visitor who is asked to add labels, graphical elements, and a topology of distinctive places in order to produce a human-friendly, *schematic map* of the environment.
4. Both, the geometric and the schematic maps, are registered one to another to relate their coordinate systems (meters and pixels, respectively). This is essential to translate pixel-related information, e.g. the visualization of the position of the robot, to geometric-related data, i.e. the (x,y) position of the robot, and vice versa.
5. At any moment, the visitor can update the built map to reflect modifications in the apartment, e.g. changes of the furniture’s layout.

Figure 4 depicts the most relevant parts of the interface we have developed to incorporate all the mapping functionalities. Note that the presented approach can be applied to any other telepresence robot with minimal changes to accommodate to its particularities.

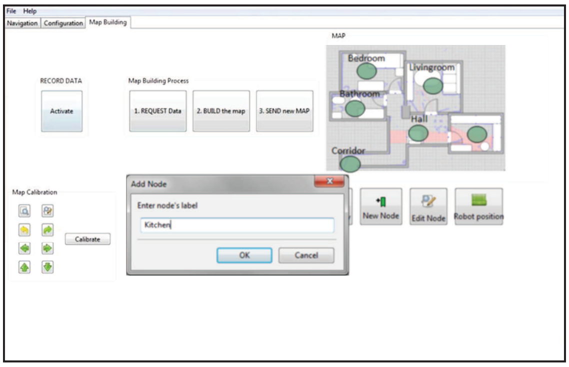
Next, each step of the map building process is described in more detail:

3.1 Recording sensorial data

The interactive mapping process is initiated by the visitor who remotely drives the Giraff robot, while scans are continuously collected. The posterior map building algorithm combines such data which may require a considerable computational effort. Given the limited computational resources of Giraff, the collected data is transmitted to the remote client to run the geometric map building algorithm. Concretely, information from the wheels’ encoders (odometry) and range data from the radial laser scanner are transmitted using the *MQ Telemetry Transport protocol* (Hunkeler et al., 2008), that is a suitable solution in mobile applications with limited resources. This protocol is based on a simple publish/subscribe fashion, especially designed for sensorial data transmission.

In our implementation we consider two messages published by the robot, i.e. `odometry` and `scan`. The

a)



b)

Figure 4: Client interface. a) Navigational view. b) Window devoted to the mapping process.

odometry message contains 2 float numbers, i.e., the odometric position (x,y) of the robot, and the scan message contains 361 integers, i.e. the distance in cm. to the closest obstacles in a range of $240°$. Messages are sent at $1Hz$., so the transmission rate is approximately 1.5 Kb/s. The client, in its turn, is subscribed to these messages and stores them until the exploration phase ends.

3.2 Geometric map building

For building a geometric map upon the received scans, the system runs an implementation of the Iterative Closest Point algorithm –ICP– (Besl and McKay, 1992) from the Mobile Robot Programming Toolkit (MRPT, 2013). ICP aims to register point-based data coming from a number of scans by finding the geometrical transformations that minimizes the square error between the registered points. This gradient descent method has been extensively used in the robotics arena, being known as *scan matching*.

Figure 5 shows an example of the resultant geometric map constructed in one of our testsites. Notice that this map, although essential for robot localization, is not appropriated for human interaction.

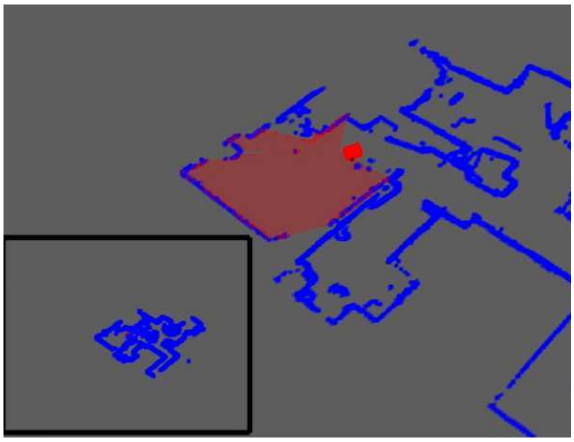


Figure 5: Example of a constructed geometric map. In red, one of the scans taken by Giraff during the map building process.

3.3 Topological and schematic map

The generated geometric map is enriched in this phase by the visitor in order to produce a suitable schematic-topological map. For that, s/he is asked to perform the following two steps:

1. Create a schematic map by adding graphical elements that represent pieces of furniture and environment structures, like doors, walls, etc., and
2. Create a topological map by selecting distinctive places, connections, and friendly names, e.g. kitchen, corridor, bedroom, etc.

While the former only aims at enhancing the visualization of the environment, the latter, i.e., the creation of a topology, including human-friendly labels, opens interesting possibilities for identifying particular rooms of the elder home and for using this high-level information as destinations for reactive navigation [1].

In the current implementation, the visitor can add distinctive places within the map by clicking on the desired point and adding an intuitive label (see figure 4b). Places, represented by nodes, can be, if desired, connected through arcs to indicate the possibility to go from one place to the other.

Regarding the schematic map, the client interface does not integrate drawing capabilities, so it requests the visitor to draw a sketch over the provided geometric one through any external drawing software, e.g. MS Visio (see figure 6). The resulting image file is then incorporated into the interface for visualization and robot commanding purposes.

[1]Although the literature normally assumes that telepresence is based on teleoperation, we extends here the convenient feature of robotic semi-autonomy.

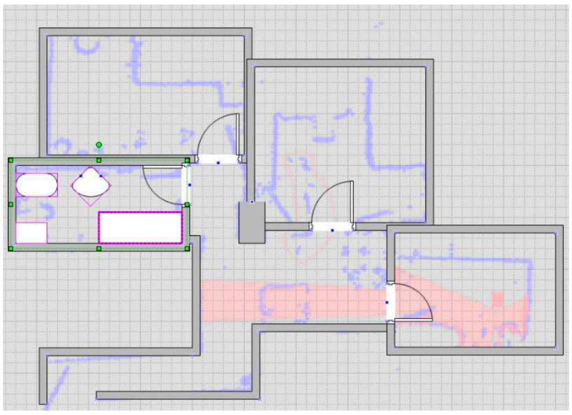


Figure 6: Example of a schematic map constructed over the geometric map utilizing external drawing tools.

3.4 Transformation between the geometric and schematic maps

To properly translate pixel-related commands, e.g. showing the localization of the robot within the schematic map, into geometrical-related information, e.g. the real (x,y) position of the robot, some transformation is needed.

The construction of the geometric map establishes the initial robot position as the geometrical coordinates' center of the map. When the ICP finalizes, and the dimensions, i.e. width and length in metres, of the apartment are known, our software generates a bitmap file and computes the pixel/metres relation for each particular environment. Given that the schematic map is constructed over the geometric one, the computed relation is kept and serves to transform robot destination points in pixels into geometrical coordinates and vice versa.

3.5 Map Update

The utility of static geometric maps is certainly limited when dealing with dynamic environments. The addition, removal or displacement of pieces of furniture may degrade the performance and accuracy of the self-localization process. For tackling this issue the visitor can update parts of the map at anytime by repeating the mapping process on a selected area. The system re-runs the ICP algorithm to create a new updated version of the geometrical map. The need of updating the map is advised by the system based on the accuracy yield by the localization module.

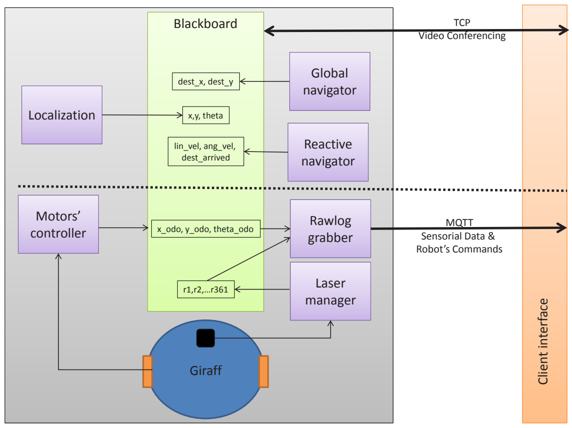


Figure 7: Software architecture. Modules from the robotic architecture share information through the blackboard. The client interface interacts with the robot by directly accessing to the blackboard and through a MQTT channel established in the rawlog-grabber component.

4 Software architecture

The software architecture considered for our map building application and the posterior usage of the constructed maps (outlined in section 5) is illustrated in figure 7. It is divided in two parts: the client interface that runs on the visitor's computer, and a robotic architecture in the side of Giraff that manages and controls its motors and sensors. All software modules have been implemented in C++, using the MRPT toolkit (MRPT, 2013).

In the client side, the interface has been implemented as a single program with two communication channels with the robot: a TCP socket for videoconferencing, and a MQTT channel for exchanging sensorial data and commanding the robot.

In the Giraff side, we rely on the OpenMORA architecture (Mapir, 2013), a particular robotic architecture based on MOOS (Newman, 2003), that considers a general, centralized blackboard from which the connected modules can share information by publishing and subscribing to particular topics. This internal communication is implemented by local TCP sockets.

The components that run on the Giraff robot can be divided into low and high-level modules. Low-level ones provide a basic access to sensors and actuators and are directly involved in the localization and mapping process. These include:

- *Motors' controller*. This module manages the Giraff motors and is in charged of establishing the desired robot velocities as well as of reading the odometry of the robot. The interaction with the blackboard is done by its subscription to these topics.

- *Laser manager*. It collects scans from the laser scanner and continuously publishes the range data of the most recent scan into the blackboard.
- *Rawlog grabber*. This module transmits the robot odometry and the collected scans published in the blackboard to the client interface using the MQTT protocol.

On the other hand, the high-level modules are software components that perform data processing for exploiting the created map. These modules are:

- *Robot Localization*. Giraff self-localization is performed by a Particle Filter technique which estimates the pose (position and orientation) within the already known map, represented as a two-dimensional occupancy grid model, through a probabilistic Bayesian framework that resembles Montecarlo simulation (Blanco et al., 2010).

  Given the limited performance of the Giraff onboard computer and the considerable computational burden of the particle filter algorithm, the localization process is executed at a low rate (2Hz) and with a reduced (but sufficient) number of particles. For visualization purposes, the pose of the robot is displayed on the map at a higher rate using the odometry positioning, which works at 20Hz.
- *Reactive navigator*. A reactive navigator automatically guides the robot to a nearby point negotiating the detected obstacles. It uses the robot pose and the sensor observations to derive the proper motors' commands to go from a point 'A' to a point 'B' negotiating any (possibly dynamic) obstacle found in the path.

  Concretely we have endowed the Giraff robot with a reactive navigation approach based on Parametrical Trajectory Generators –PTG– that has successfully proved its performance and reliability in cluttered spaces (Blanco et al., 2008).

  In short, the underlying idea of the PTG-based reactive navigator is to abstract both the geometry of feasible paths and the robot shape into a space transformation, in such a way that simpler obstacle avoidance methods (designed to deal with circular, holonomic robots) can be used to determine the next robot movement into such transformed space.
- *Global path planner*. This module uses the topological map created by the user to search for a path from the current position of the robot to the destination given by the user in terms of labels, e.g. "kitchen", "livingroom", etc. The global path planner complements the reactive navigation which is not appropriated for far destinations, since it only takes into account the current perception of the robot. In contrast, the global navigator exploits the topological map enabling the user to choose a destination through its label. The global navigator executes an A* algorithm (Hart et al., 1968) to search the shortest path to the goal in the created topology, producing a sequence of nodes, i.e. distinctive places, connected by arcs. Each node stores the geometrical position, (x,y), of the place in the coordinate system of the robot, and are sequentially sent to a reactive navigator, which is fed with the geometrical position of the next node of the path until the destination is reached.

5 Discussion and Conclusions

Enhancing the teleoperation interface with maps brings a number of advantages for the robot driver. On the one hand s/he can benefits from a certain degree of navigational autonomy which explicitly requires some type of world representation. Although telepresence implies the continuous and effective participation of a human controlling the robot, providing certain automatic maneuvering can be desirable. For instance when a driver wants to traverse long corridors or pass through narrow spaces, s/he would prefer to delegate these bored and unpleasant tasks directly to the robot. This leads to a reduction of the mental attention and workload of the visitor who can focus on the social or professional communication which is the ultimate aim of a telepresence robot. For exploiting this feature, the visitor should be able to select a nearby destination in any representation of the space, arising thus the need of a convenient map. Moreover, apart from relying on a reactive navigator to relieve the visitor from maneuvering, the use of a topological map is required to also enable him to establish a global, distant destination given in terms of friendly, well-known labels, e.g. kitchen.

On the other hand, having a graphical representation of the real time position of the robot within a schematic map of a house is especially useful for the visitor to facilitate the teleoperation and eliminating her/his very likely disorientation.

These remarks motivate the need of having a convenient representation of the environment for robotic telepresence applications. In this paper, we have described a map building process that builds upon well-known robotic techniques, and a graphical interface that permits the visitor to remotely construct and exploit the map in the terms aforementioned. The result has been tested in several testsites in Spain with the Giraff telepresence robot proving the suitability of our approach for this type of applications.

Our short-term research aims at providing dependability to the system by incorporating a RGB-D camera (Kinect-like) which helps in the localization and obstacle detection tasks.

ACKNOWLEDGEMENTS

This work has been supported by two projects: the EXCITE project, funded by AAL (Ambient Assisted Living) Program and Instituto de Salud Carlos III, and by GiraffPlus: Combining social interaction and long term monitoring for promoting independent living, funded by the European Community's Framework Programme Seven (FP7) under contract #288173. FP7 - ICT - Challenge 5: ICT for Health, Ageing Well, Inclusion and Governance.

REFERENCES

Besl, P. J. and McKay, N. D. (1992). A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256.

Blanco, J.-L., González-Jiménez, J., and Fernández-Madrigal, J.-A. (2008). Extending obstacle avoidance methods through multiple parameter-space transformations. *Autonomous Robots*, 24(1):29–48.

Blanco, J.-L., González-Jiménez, J., and Fernández-Madrigal, J.-A. (2010). Optimal filtering for non-parametric observation models: Applications to localization and slam. *The International Journal of Robotics Research (IJRR)*, 29(14).

Coradeschi, S., Kristoffersson, A., Loufti, A., Rump, S. V., Cesta, A., Cortellessa, G., and González-Jiménez, J. (2011). Towards a methodology for longitudinal evaluation of social robotic telepresence for elderly. 1st Workshop on Social Robotic Telepresence, held at HRI 2011.

Giraff (2013). Giraff A.B. Technologies. http://www.giraff.org/.

González-Jiménez, J., Galindo, C., and Ruiz-Sarmiento, J. R. (2012). Technical improvements of the giraff telepresence robot based on users evaluation. In *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*.

Hart, P., Nilsson, N., and Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *Systems Science and Cybernetics, IEEE Transactions on*, 4(2):100 –107.

Hokuyo (2013). Hokuyo homepage. http://www.hokuyo-aut.jp.

Hunkeler, U., Truong, H. L., and Stanford-Clark, A. (2008). Mqtt-s - a publish/subscribe protocol for wireless sensor networks. In *COMSWARE*, pages 791–798. IEEE.

Mapir (2013). Mapir homepage. http://mapir.isa.uma.es.

MRPT (2013). The Mobile Robotic Programming Toolkit (MRPT) homepage. http://www.mrpt.org.

Newman, P. M. (2003). Moos - a mission oriented operating suite. Technical Report OE2003-07, MIT Dept. of Ocean Engineering.

Tsui, K. M., Von Rump, S., Ishiguro, H., Takayama, L., and Vicars, P. N. (2012). Robots in the loop: telepresence robots in everyday life. In *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, HRI '12, pages 317–318, New York, NY, USA. ACM.