BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2018
http://excel.fit.vutbr.cz

Indoor Navigation System on Mobile Device

Peter Žiška

Abstract
This work is focused to suggest an approach, which simplifies creation of indoor navigation systems on mobile devices. It means that I introduce support tool where user can scan useful data and generate navigation representation. Localization system is implemented with wifi fingerprinting method, which is divided into two phases: offline – creating fingerprinting database and online – comparing data with fingerprinting database. Offline phase is complicated in term of time, because it is needed to manually build database from whole building space. To solve time problem the scanning tool was developed. Navigation is represented with bi-directional weighted graph. The graph is partially generated based on geometric representation of building and with help of user, graph is finished by adding some missing vertices of graph. All you need to implement localization and navigation application is geometric representation of building and this tool, which generates graph and scans for access points. Approach reduces time to implement indoor positioning system based on WIFI fingerprinting method and creates navigation graphs.

Keywords: Navigation — Localization — WIFI — Fingerprinting — Android — Graph Generation

Supplementary Material: Demonstration Video — Downloadable Code

*xziska02@stud.fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

Creating maps for indoor localization and navigation is expensive and time consuming process, but for other purposes are sufficient different accuracies. My approach uses simple building definitions with coordinates of every office to implement map. The process of obtaining fingerprints database is implemented alongside generation of navigation from map representation in support tool. Output of the tool is then used in mobile application for localization and navigation purposes.

In the implementation of indoor navigation system there are two main problems – localization and navigation. Indoor localization is difficult, because the GPS could not be used for it's permeability through the walls. There is a way to use different signals such as WIFI or Bluetooth but implementation could be time consuming. Second problematic part is navigation. Its job is to find the shortest path to target place. To find such path, I need to think about navigation representation. It could be also time consuming to create navigation graph manually, so I would like to find easier way, which will remove manual insertion of graph.

Location determination indoors and representation of navigational paths are well known problems. Different approaches were discussed using WIFI or Bluetooth signals, picture recognition or dead-reckoning for indoor localizations methods. Navigation graphs are usually represented by bi-directional graphs, but differences between other solutions are in the generation process of this structure.

In [1] there is proposed indoor localization system based on recognition of panoramic pictures of building points of interest (PoI). They divided process of building navigation system into two steps – first step was to generate map by taking panoramic pictures in some places of building. Panoramic pictures are used to implement localization, while user choose his location based on panoramic pictures. Also pictures are used to provide navigation graphs, where PoI are vertices of graph. For determining movement of user was used dead–reckoning method with usage of accelerometers and magnetic field. To find shortest path they used Dijkstra's algorithm. This approach matters on user's engagement in term of determining his location. It could cause problems for users to choose exact position, if there are plenty of recorded panoramic pictures. On the other hand if there are only few pictures taken, user will be limited only for these pictures and if he will be between two points he will need to choose one and it is automatically the wrong one. In this work was suggested the improvement with real time comparison of pictures could be helpful.

Another solution [2] uses also dead–reckoning for determination of user's heading. To be able to use dead–reckoning method, user has to find its location manually by clicking on exact position. Map and navigation representation was obtained by OpenStreetMap platform[1], where creator of map can define geographical location of building and manually defines possible paths for navigation process. OpenStreetMaps are good solution to define map but the process of building navigation can be time consuming. They focus on implement accurate dead–reckoning method.

The most complex approach and implementation of the selected works was suggested in [3], where they used WIFI fingerprinting method as localization method along with dead–reckoning method to real time positioning. Fingerprints for the method was collected by the robot. Robot automatically scans and saves the fingerprints to database. Implementation of navigation system is based on $D*$ algorithm, which will find the shortest path starting point to destination by going through the map representation pixel by pixel. Algorithm recognizes walls as the color of pixel changes and it will go to another one until it reaches the target.

My intention is to implement the tool, which will run as desktop application. With this tool user could both generate graph and scan the data from access points. Storing data into database from access points especially signal strengths will be provided by user's clickable event on the indoor space, which will be drawn from geometric representation of building space. Input of the tool is geometric representation – room's coordinates of every corner and the representation will be used to compute navigational graph. Navigational graph will be based on visibility graphs and the weights of every edge will be computed as the Euclidean distance from vertices which form the edge. After creating necessary files (fingerprint database and graph representation), the approach will be demonstrated on mobile application.

In this work, I propose the solution based on modern techniques (described in section 2), design steps to overcome practical problems and demonstrate the applicability of my solution in a set of working tools and a mobile application (described in section 3).

[1] OpenStreetMap – https://wiki.openstreetmap.org/wiki/Main_Page

2. Theoretical background

2.1 Localization technique

Nowadays, user's location is determined by Global Positioning System (GPS) in the outdoor space. However GPS has a big disadvantage indoors because of permeability of satellite signals through the walls of building and that causes unexpected localization. Here comes the possibilities of using different approaches with other signals such as WIFI, Bluetooth or Radio Frequency Identification (RFID).

Using WIFI access points with its development and great availability could be a great option for replacing GPS in indoor space. For localization with access points there are many methods for implementing indoor localization from trilateration (also GPS uses this method) to triangulation or wifi fingerprinting method. These methods can be developed for every system, which transmits radio signals. I will present only WIFI fingerprinting method, because this method is used for determining location in my approach.

WIFI fingerprinting

WIFI fingerprinting [4] is divided into two parts. In the first part – offline phase, the point map is built. Map point is represented by the vector of signal strengths of all available access points from the spot in the map. Then, fingerprints are stored into database for later comparison in the online phase of the localization process.

In the online phase user would like to determine his location with mobile device that is running scanning algorithm for collection of online fingerprint. After the fingerprint is collected, it is used in a comparative process with the database created in offline phase. The comparison can take place based on the interval selected according to the signal strengths from the online phase and further searching for the signal strengths from the database corresponding to the selected interval. The record with the highest number of signal strengths corresponding to the intervals is declared as the user's position.

Another option to compare database and online fingerprint is usage of metrics. Most often the Euclidean distance between reference points from mobile device and records from database is used. The record with the shortest distance is declared as user's location in the map. Computing of distance is based on formula 1, where d_i is the distance of fingerprint and actual position, $|A|$ is the number of access points in the fingerprint, R (online fingerprint) and R^* (offline fingerprint) are signal values for same access point. Values of signal strengths are measured by decibel-milliwatt [dBm]. Example of map with fingerprints is shown in the figure 2, where dots represents fingerprints.

$$d_i = \sqrt{\sum_{k=0}^{|A|} (R_k - R_k^*)^2} \qquad (1)$$

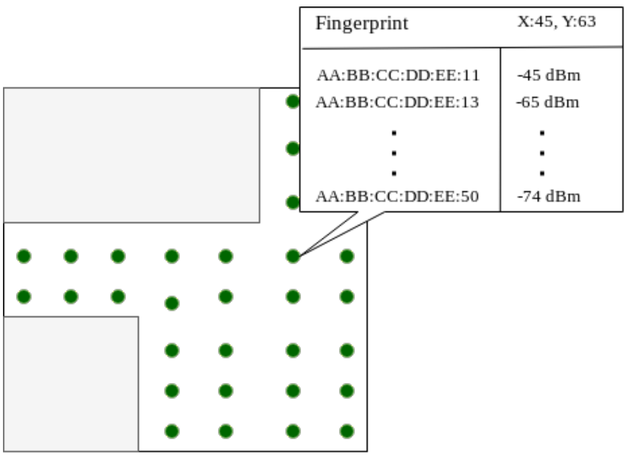

Figure 1. Fingerprinting database created for a building space. Every green dot represents a vector of surrounding MAC addresses from access points with correspondent signal strengths.

2.2 Navigation representation and construction

The purpose of a navigation system is to find a path from starting point to target point. Path, that was suggested by navigation system should be the shortest to save some time for user. The navigation gains the starting point from localization techniques mentioned above. Target point is defined by user from their input. The key of finding the shortest path lies in representation of the navigation. Probably the most used representation is the graph, which has edges with cost. In such structure there could be easily found the shortest path with algorithms such as Dijsktra's algorithm or A*.

Graph

Conceptually is a graph [5] defined with vertices and edges between single two vertices. Formally, a graph is a pair of sets (V,E), where $V = \{v_1, \ldots, v_n\}$ is a nonempty set of all vertices in the graph and E is a set of edges, formed by pair of vertices. Edge starts with some vertice and ends in one, but there could be exception where edges starts and ends in same vertice – such edge is called loop.

Visibility graph

Visibility graphs [6] deal with objects, which can create natural obstacle in the path between vertices. Obstacles are defined in a polygonal representation. Visibility graphs are formed by vertices and edges, which is defined by pair of vertices, which are not interrupted by a obstacle object. To find out these pairs it should be tested for every point between pair and when the line is not interrupted by obstacle the edge could be created. Example of visibility graph is described in the picture 2.

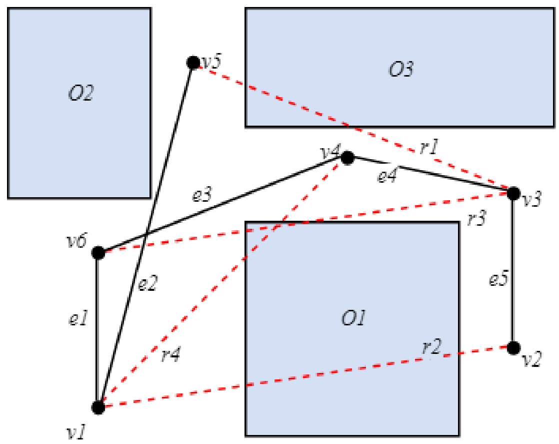

Figure 2. Visibility graph example. There are three obstacles `O1, O2` and `O3`. Graph contains six vertices `v1, v2, v3, v4, v5 and v6`, between them are created multiple edges `e1, e2, e3, e4 and e5`, that meet visibility graph conditions.

3. Experiments and implementation

The solution I propose consists of two phases. The map generation phase is a one time process and is performed first for each indoor location. Generated map consists with localization database of fingerprints and navigation graph defined in XML files. The Localization and Navigation phase will use the map generated in the first phase for user navigation in mobile application. All inputs and outputs of support tool and mobile application are shown in figure 3.

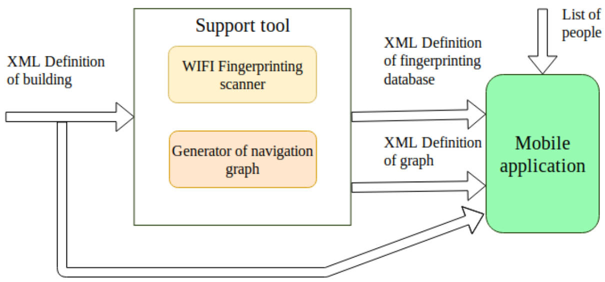

Figure 3. System definition divided into two parts – support tool, where user scans for fingerprints and generates navigation graph and mobile application with input from the tool.

3.1 Creating fingerprint map

Implementation of WIFI fingerprinting method is divided into two phases as mentioned in section 2. The crucial phase is the offline phase of this method, where the database is created. For simplification of the collection process, the support tool for scanning fingerprints was developed.

Input of the support tool is the geometric representation of building floors defined in XML file. The representation is used for drawing floor map on a window and it is used for user's interaction. The user chooses place where he stands in the map and after the click on chosen place, the algorithm of collecting signal strengths with corresponding access points will start. Algorithm uses system tool *iwlist* to collect the signal strengths from surrounding access points to create fingerprint. This tool is repeated 4 times with one second gaps in a script for getting better and more accurate data, where all signal strengths for corresponding access points are averaged into the fingerprint. When user walks through whole building with scanning, he can save his records into XML file, which will be used in the mobile application as the source of fingerprinting database.

3.2 Generation of graph

For generation of graph I use geometric representation of rooms. Geometric representation means, that coordinates of every room's corners will be stored and used. With such representation I can define neighboring relation with every room and the relation can be calculated by testing, that room's lines are formed by points, which can be tested against other rooms, if these points are located in the other room. I want to find only relation between halls and rooms, where algorithm goes through every room's line and mark the first occurrence of containing point in the hall and the last occurrence of point in the hall. These two points are used to compute terminal vertice (doors) by averaging their coordinates. Another necessary vertice is the main vertice. The main vertices are computed easily only for halls because I want to walk only through halls by averaging hall's coordinates.

After generating these two types of vertices, algorithm can connect them and create edges. Connections are created, as mentioned, between main vertices and terminal vertices, while terminal vertices are connected only with the main vertice, the closest one. With this approach, the graph I get is not well connected because graph has few main vertices. To solve this problem user can define another vertices by clickable event on map, that will be automatically considered as the main vertice. After insertion, all edges in the clicked room will be recalculated.

Main vertices are connected with each other. This connection will produce a lot of useless and redundant edges. Finding the shortest path will be time consuming so I need to reduce them. Reduction algorithm is based on finding another path of edges to destination vertice of potential reduced edge. When such path exists and at the same time every edge in this path is shorter than potential reduced edge, this edge is removed from graph representation.

If user is not satisfied by graph's edges, they can be added with choosing two vertices and after that, edge will be created and saved to graph. In the picture number 4 there is described process of computing graph for indoor navigation system using geometric representation of building plans.

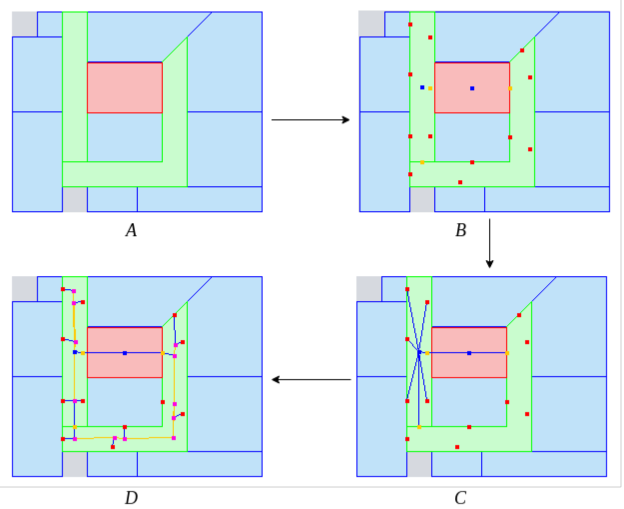

Figure 4. Process of building graph for navigation purposes. There are 4 phases of building *(A, B, C, D)*. Phase *A* shows building representation (Classic rooms are marked with blue color, with green halls and finally with red color stairs or lifts). On phase *B* you can see generated vertices. Phase *C* shows automatic connection between main vertices and terminal vertices. On phase *D* you can see added vertices by user and recalculation of edges with reduction.

Final step is to connect every floor of the map. Connection begins by determinating which floors are above each other. When program gets input map with specified floor numbers, numbers are sorted and program knows the basics. After that, algorithm will start from the bottom of the building (lowest floor) and continue with floor above. All terminal's vertices of lower floor will be tested with terminals of above floor, so if one of vertices is contained in above terminal, the edge will be created with this vertice and the one of main vertices from terminal.

After the process of generation navigation graph, the structure of graph is saved into XML file for every floor in building. XML file defines vertices and edges between them and it is used to navigate user in mobile application.

3.3 Mobile application

Implemented tool is returning graph representation and fingerprinting database defined in XML files. These files are then imported into the mobile application and with the geometric representation of building, application is easily provided with new information for displaying indoor floor map. Application implements online phase of WIFI fingerprinting method. To determine direction of user's look, application uses inertial sensors, especially accelerometers and compass, that help it to compute the direction. The navigation files determines possible paths in the map and with the building representation user can easily find the target of his heading by choosing or searching for a person or an office. Final implementation of application and its user interface are shown in the figure 5. In this picture you can see the navigational path, that is represented by graph generated with support tool. Path is searched with Dijkstra's algorithm.

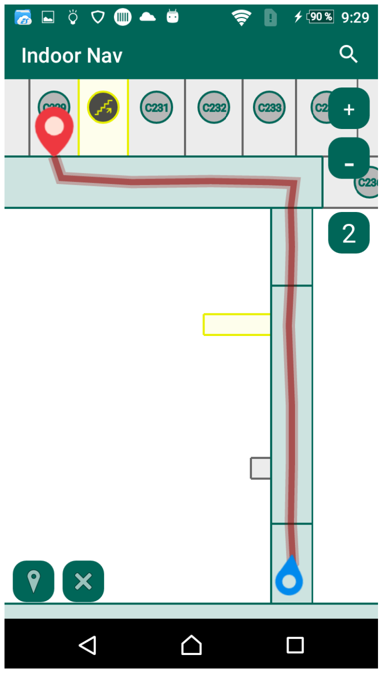

Figure 5. Visualization of mobile application, where the current position, the direction of user and the shortest path to his target place are shown.

Mobile application was implemented to localize a device in buildings. I demonstrate the localization process in the picture number 6 in a hall inside the Faculty of Information Technologies. I walked through the hall with mobile application on and the application was computing my position. On my way through the hall I got fingerprints, which was the best match for my current position and after that I stored coordinates of matched fingerprint. Picture visualizes good results, but tests shows few places especially near the stairs, where localization algorithm returns wrong floor number and it could be confusing for user. Possible mistakes in localization can be caused by time or building dynamic life, because when these conditions change, the signal strengths of fingerprints will change too.

4. Conclusions

The aim of this work is to propose approach, which will create efficiently localization database for WIFI fingerprinting and graph for navigation in the indoor space. To demonstrate suggested approach, the mobile application was developed. Localization system was developed by using WIFI fingerprinting method.

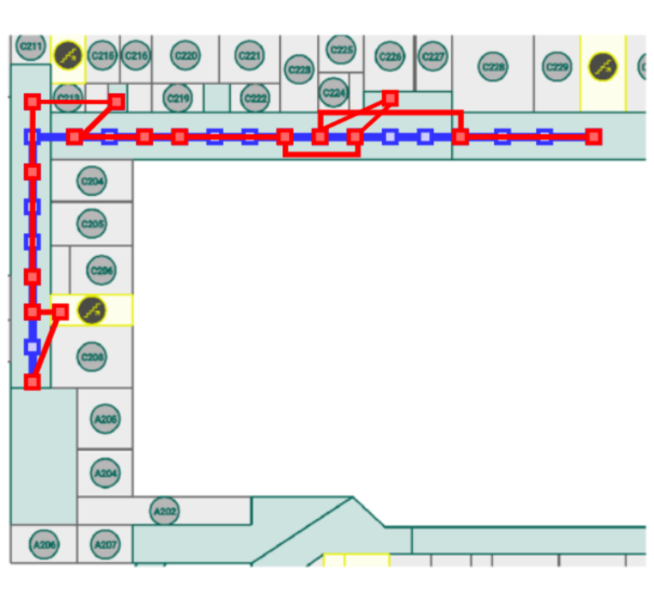

Figure 6. Demonstration of the localization process, where blue path represents walking path and with red color is drawn localization path.

WIFI fingerprinting method needs the fingerprinting database, to create this database I implemented a tool to collect signal strength informations and user can easily construct navigation graphs with this tool too. The output of support tool is then used in mobile application to implement navigation system.

The approach of scanning indoor space was suggested to create database for WIFI fingerprinting indoor localization method. Navigation system is represented by graph, that is generated by geometric relations between rooms in the building.

Probably the biggest problem of suggested and implemented approach is portability of scanning indoor space for fingerprints. Solution lies in mobile application, which will lead to easier way of scanning space. Another improvement could be used to fully automatically generate navigation graphs with the skeletonization of halls. Skeletonization will give us skelet of halls, where vertices of rooms can be directly connected to the nearest point from the skeleton.

Acknowledgements

I would like to thank my supervisor Ing. Vítězslav Beran, Ph.D. for sharing his knowledge and ideas.

References

[1] Shivam Verma, Rohit Omanwar, V Sreejith, and G S Meera. *A Smartphone Based Indoor Navigation System*. IEEE, 2016. [Online].

[2] Jó Agila Bitsch Link, Paul Smith, Nicolai Viol, and Klaus Wehrle. *FootPath: Accurate Map-based Indoor Navigation Using Smartphones*. RWTH Aachen University/ComSys, Aachen, Germany, 2011. [Online].

[3] Felipe Meneguzzi, Balajee Kannan, Katia Sycara, Chet Gnegy, Evan Glasgow, Piotr Yordanov, and M. Bernardine Dias. *Predictive Indoor Navigation using Commercial Smart-phones*. Carnegie Mellon University,Pittsburgh, USA, 2011. [Online].

[4] Eduardo Navarro, Benjamin Peuker, and Michael Quan. *1 Wi-Fi Localization Using RSSI Fingerprinting*. California Polytechnic State University, USA, 2011. [Online].

[5] Daniel Turzík and Pavla Pavlíková. *Diskrétni matematika*. VŠCHT Praha, 2007.

[6] M de Berg, O Cheong, M van Kreveld, and M Overmars. *Computational Geometry Algorithms and Applications*. Springer, 2008.