Självständigt arbete på grundnivå

Independent degree project - first cycle

Datateknik
Computer Engineering

visualization tool used in indoor positioning system

Huan Huang
Abstract

The necessity and usefulness of indoor location technique increasing significantly. It can achieve the function that locate people and goods. But the indoor location technique need to delivery a lot of message to user. There is not software which can fully meet these functions. In order to allow users receive the message quickly and precisely, it is very significant to design an user-friendly visualization tool. In order to improve the efficiency of software reuse, this project also develop each function module into component. This visualization tool can show the real position and estimated position of objects and show error between real position and estimated position in different angles to user. So user can improve the quality of indoor position system based on the error.

This project is developed from user’s need: use QT as a development framework, design each component, complete the design and implementation of this tool.

Keywords: Visualization, QT, Human-computer-interaction
Acknowledgements / Foreword

First, I want to thank my supervisor, tingting zhang professor because of her guidance and help.

Also, I want to thank my friend who give me help during thesis work and presentation.
# Table of Contents

Abstract .................................................................................................................................................... iii
Acknowledgements / Foreword ...................................................................................................................... iv
Terminology / Notation ............................................................................................................................... vii

1 Introduction ........................................................................................................................................ 1
   1.1 Background and problem motivation .............................................................................................. 1
   1.2 Overall aim / High-level problem statement .................................................................................. 1
   1.3 Scope ............................................................................................................................................. 2
   1.4 Concrete and verifiable goals / Detailed problem statement ......................................................... 2
   1.5 Outline ......................................................................................................................................... 2

2 Theory / Related work ............................................................................................................................. 3
   2.1 QT .................................................................................................................................................. 3
   2.2 Visualization ................................................................................................................................. 3
   2.3 Eigen ............................................................................................................................................. 3
   2.4 Indoor location system .................................................................................................................... 4
   2.5 Software component technology .................................................................................................... 4

3 Methodology / Model ............................................................................................................................... 5

4 Implementation ..................................................................................................................................... 6
   4.1 The main interface .......................................................................................................................... 6
   4.2 Input parameter component ............................................................................................................ 7
   4.3 Map component ............................................................................................................................. 7
   4.4 Select symbols and colors component ........................................................................................... 7
   4.5 Move unit component ...................................................................................................................... 7
   4.6 Show error component ................................................................................................................... 7
   4.7 Show information component ......................................................................................................... 8

5 Results .................................................................................................................................................. 9
   5.1 Main interface ............................................................................................................................... 9
   5.2 Create scene and open a scene ....................................................................................................... 9
   5.3 Move reference unit ....................................................................................................................... 15
   5.4 Show error .................................................................................................................................. 15
   5.5 Select symbols and colors ............................................................................................................ 17

6 Conclusions / Discussion ........................................................................................................................ 18
   6.1 Future work ................................................................................................................................. 18
   6.2 Ethical consideration .................................................................................................................... 18

References ................................................................................................................................................ 20
Terminology / Notation

(Choose one of the headline alternatives.) A possible list of terms, abbreviations and variable names with brief explanations may be placed after the table of contents, but is not required. Note that although a term is explained in the list of terms, it should also be explained in the chapter text where it is used the first time.

**Acronyms/Abbreviations**

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMU</td>
<td>Real Mobile unit</td>
</tr>
<tr>
<td>EMU</td>
<td>Estimated Mobile Unit</td>
</tr>
<tr>
<td>RU</td>
<td>Reference Unit</td>
</tr>
</tbody>
</table>
1 Introduction

1.1 Background and problem motivation

Indoor location technique is used to provide location information of people and goods. Indoor location technique is very practical and has large space to expand. It can be used in complex environment. For example, library, stadium, warehouse, mine, etc. It can improve the quality of life. It needs to show complex, comprehensive location information to user. Original information processing methods, such as digital signal processing, digital image processing, can not fully meet the multifarious needs of users. There is not software for visualizing the data collected from indoor position system. Although people can use other software, for example, matlab and excel, to convert the data into graphics or images, these software can only provide limited function and can not meet different needs of user. And because these software do not provide enough interface to user, it will be difficult and complex for common users when they want to set up table or image to form what they like. It is important to show the error between the real position and estimated position to user, so user can improve the system. In order to show the information precisely, intuitively, it is important to design an user-friendly, visualization tool.

First, This project is developed by QT. Qt is a cross-platform application development framework, so it can provide wide support for object-oriented application. Second, This project uses software component technology to develop the components of visualization tool, so it is extendable. Third, the visualization tool provide different ways to show information, so it is flexible and it can meet different people’s preference and deliver information to user efficiently.

1.2 Overall aim / High-level problem statement

Visualization of indoor positioning system faces some problems. It needs to show complex information to user: a lot of real position and estiamted position and their relation and information of certain unit.

The project is to present visualization work of position system. It helps people understand the location information and other related information. This project also develop each function module into component. It’s component can be reused. It will reduce development time for later developers. Also, It is easy to extend this software.
1.3 **Scope**

The study has its focus on visualization tool. It will design a simple but practical interface and provide flexible ways to show information. It also focuses on software component technology. It can improve the efficiency of later related software development.

1.4 **Concrete and verifiable goals / Detailed problem statement**

This project is to design a software that allows visualization of the position system. This project can be achieved by completing these goals:

1. Study the theory of visualization and choose the suitable development framework to develop this software.

2. Design a simple, but practical UI of this software.

3. Learn about software component technology and design each component of this project.

4. Design some ways to show information.

5. Evaluating the result and propose the future work.

1.5 **Outline**

Chapter 1 introduces project’s background and main content of this paper. Chapter 2 describes the theory and technology of this project, present some related work. Chapter 3 introduces the methodology used to achieve the goal mentioned in chapter 1.4. Chapter 4 describes how this program is implemented. It also introduces some ways to show the error between real mobile units and estimated mobile units. Chapter 5 introduces the result of running the program. Chapter 6 compares the running result and expected result. It also proposes the drawback of this project and future work.
2 Theory / Related work

2.1 QT

QT is a cross-platform C++ GUI application development framework. QT is widely used for developing application in various software and hardware platform. Qt uses standard C++ with extensions including signals and slots that simplifies handling of events, and this helps in development of GUI which receive their own set of event information and should process them accordingly[1]. There are some advantages using Qt: excellent cross-platform features, Object-Oriented, rich API, support 2D / 3D graphics rendering.

2.2 Visualization

With the social development, human activities show the characteristics of spatial expansion and time transience. Accordingly, a large amount of information generated. Visualization are theory, methods and techniques using the theory of computer graphics and image processing technology, to convert the data into graphics or images displayed on the screen and interact with users. Visualization has been an effective way to communicate both abstract and concrete ideas[2]. Practical application of information visualization in computer programs involves selecting, transforming, and representing abstract data in a form that facilitates human interaction for exploration and understanding[3]. The use of visualization presenting information is not a new phenomenon. It has been used in maps, scientific drawings, and data plots for over a thousand years. The recent emphasis on visualization started in 1987 with the publication of Visualization in Scientific Computing. Visualization today has ever-expanding applications in science, education, engineering (e.g., product visualization), interactive multimedia, medicine, etc.

2.3 Eigen

Eigen is a high-level C++ library of template headers for linear algebra, matrix and vector operations, numerical solvers and related algorithm sand it is often noted for its elegant API, versatile fixed and dynamic matrix capabilities and a range of dense and sparse solvers[4]. Eigen is a very powerful library. It is very convenient to use it to do insert, delete, modify and arithmetic operation in matrix.

2.4 Indoor location system

GPS is not suitable for locating indoor people or objects, because the roof and walls will attenuate the signal. So it is important to find a solution to locate indoor objects. Indoor location system is a good choice. An indoor positioning system (IPS) is a solution to locate objects or people inside a building using radio waves, magnetic fields, acoustic signals, or other sensory information
collected by mobile devices[5]. Indoor locating which uses electromagnetic waves from indoor transmitters to indoor receivers can degrade the effect of signal attenuation. Indoor location system becomes very popular recent years. The position information is very useful. A network is formed by using different communication devices equipped with communication technologies in various place. Location-base service need to be developed in this network.

2.5 Software component technology

An individual software component is a software package, a web service, a web resource, or a module that encapsulates a set of related functions[6]. Components are cohesive and modular. The interface of component can be seen as a signature of this component. User don’t need to know the inner structure of this component and can use it directly. Component is substitutable. It can be replace by another component. For example, component B can replace component A, if component B provided at least function that component A has provided and component B don’t use the function more than component A. Reusability is an important attribute of software component. One important purpose of designing components is allow other programmers reuse these components. Reusability of component helps programmers focus on more complex function and programmers don’t need to take a lot of time in basic component. Programmer doesn’t need to know how the component works. He just need to know how the integrate component with application, just like a driver does not need to how a engine work. So programmer can concentrate on the main function of application rather than on basic component. Software component technology helps to reuse component again and again, which is time and cost saving. When these components are used again and again, these components become more reliable.
3 Methodology / Model

This project is developing a visualization tool for position system. This tool should be user-friendly, flexible, extendable.

In order to understand background and research status of this project, the first thing is searching some papers, articles and books about visualization on the Internet and learn about the theory of visualization. After this, compare different development framework which aim at developing UI for software to choose an development framework for this project. Next thing is to design an interface to show the data. In order to design an user-friendly interface, learn about the user’s demand and observe how other software design their interface.

After designing the interface, learn about software component technology and design each component of this project. Each component has its own function. The combination of these components can compose a visualization tool.

The next thing is to visualize the data provided by user. This program should be flexible and extendable, so it should provide different ways of showing data to allow user to choose and user can choose the way they like. So, it is important to read some papers, articles for learning ways of showing information and choose some suitable ways for this project.

Test and evaluate this program is also important. In order to achieve this goal, it is important to check whether the data is show as expected and each component can achieve their own function. It should also visualize the data from different position systems. The future work should be proposed. So, the drawback and further needs of users should be found.
4 Implementation

This chapter explains the structure of project and each function. It describes function and analyses of each component.

4.1 The main interface

This project is constructed the visualization for the position system. According to user’s need, I need to show the units on the map and show the information of certain units the user want to choose. I should also draw the legend of map to allow user get a better understanding of what the map draw. So, this program divide the UI into three parts. The first part is main part of UI and it is used for showing the map. On this part, I show the map user selected, and then show references units, real mobile units, estimated mobile units on the map. The second part is to show the information of units which user choose. It is on the right side of UI. The third part is on the bottom of UI. It is for showing the legend of map. Sometimes, user want to change sizes between different parts. For example, when the user focus on the map, he/she hope the map part become bigger and the legend part and information part become smaller. I use splitter in QT to achieve this function. User can change the size of each part by drag the boundary of this part. The UI should also contain the menu bar and status bar for providing function buttons to user.
4.2 Input parameter component

In order to create a scene, user need to input some parameters of the scene. Input parameter Component is a component that allow user input parameters of this scene. The program will show the map, reference units, real mobile units, estimated mobile units, legend of the map in the map area. So the program need to know the directory of map, coordinates of these units. And then based on the directory and coordinates, the program show these units on the map. In order to transform the coordinate in real work to coordinate in QT, user also have to input the the range of map in horizontal direction and vertical direction. Program can use the size of pixel and the size of map to transform. Program can save these parameters into a txt file. When user want to open the scene again, he/she just need to open the txt file and program will use these parameters to construct the scene again.

4.3 Map component

Map Component is a main component of this program. The Map will be show in this component and the show the reference units, real mobile units, estimated mobile units on the map based on the coordinates user provided. This component also need to draw the legend of map. At the beginning, this component uses default symbol and color to draw the legend. If user uses other colors and symbols to represent units, the legend of map will change at the same time. Communication between map component and select symbol and color component is achieved by using signal and slot mechanism.

4.4 Select symbols and colors component

At the beginning, the program will use default symbols and colors to represent reference units, real mobile units, estimated mobile units. In order to allow user choose the symbols and colors they like, this component provide some basic symbols and colors in the selection box. The signal and slot mechanism can be used in communication between these two components. Select symbols and colors component sends a signal to map component and map component use the symbols and colors to represent reference units, real mobile units and estimated units and change the symbols and colors in map component.

4.5 Move unit component

Sometimes, user need to move the reference unit on the map and change coordinate in the file. So this program should provide this function. This function can be achieved by Move Unit Component. User select the number of reference unit he want to move. And the reference number will become bigger to allow user confirm that it is the reference unit he want to move. If user confirm, program will move the reference unit to new place on the map.
4.6 **Show error component**

Show error component is a component that provide the function that show the error between the estimated mobile units and real mobile units. In order to meet different users’ preferences, this component should provide different ways to show the error. The first way is using graph to show error. The second way is using table to show error. The third way is using pie and table to show error.

4.7 **Show information component**

Sometimes, user needs to look up information of certain unit. Show information component can achieve this function. There is selection box in the show information component. When a unit is selected, program will retrieve the information of this unit and show these information in information area. The second way to show the error is clicking the unit in map component. After click, the map component will send a signal to show information component. Show information component captures the signal and show the information based on the signal.
5 Results

5.1 Main interface

Figure 1, main interface

Figure 1 is the main UI of this program. Top of the main UI is the menu bar. The first menu in menu bar is “File” menu, which include create a scene, open a scene, exit the program actions. The second menu in menu bar is “Function” menu. It includes two functions, “move the reference unit” and “show error”. The third menu is “Help” menu. User can find the information about how to operate this software in this menu. The status bar is below the menu bar. It includes “New File”, “Open File”, “move reference unit”, “show error” shortcut button. The UI below status bar is divided into three parts. The main part is for showing map, reference units, real mobile units, estimated mobile units. The bottom part is for showing the legend of map. The right part is for showing the information of reference units, real mobile units, and estimated mobile units.

5.2 Create scene and open a scene
The name edit box is for user to input the name of scene. Because the program will record these parameters into a file, the file needs a file name. And the name inputted in edit box will be the name of file. If the name is the same with former file, this program will pop up a warning dialog as figure 3. If you click the “Yes” button, the new file will replace the former file. And then the user should select the input type. After selecting the input type, user need to input the range of map in horizontal and vertical direction. The Load map is for user to choose the map he want to show. After clicking this button, it will show a file selection dialog as figure 4. User can choose the map in this dialog and the program will record the directory of the chosen map. The “Load RU” button is for loading the file which have the data of reference units. After clicking this button, it will show a file selection dialog as figure 5 to choose the file contained reference units’ data. The “Load RMU” button is for loading real mobile units file contained real mobile units’ data. After clicking this button, the program will show a file selection dialog as figure 6 to allow user choose real mobile units file. The “Load EMU” button is for loading estimated mobile units file contained estimated mobile units’ data. After clicking this button, the program will show a file selection dialog as figure 7. And then the user click the ok button, the program will show the data in files user has chosen. If user click “cancel” button, the program close the dialog and do nothing.
Visualization tool used in indoor positioning system

Huan Huang

2015-06-05

Figure 3, Warning dialog

Figure 4, Open image dialog
Figure 5, Select Reference unit dialog
Visualization tool used in indoor positioning system
Huan Huang

2015-06-05

Figure 6, Select real mobile unit dialog
Figure 7, Select estimated mobile unit

The second button in the status bar is “open file” button, which can open the scene user has created. After clicking this button, it will show a file selection dialog as figure 8. User can choose file which record the parameters of a certain scene in this dialog. The program will read the parameters in the file and use these parameters to show the scene again, just as figure 9. After open the scene, user can loop up the information of certain point. There are two ways to look up the information of a certain unit. The first way is to click the unit on the map. And then the program will show the information on the right side. When some points are close, it is inconvenient for user to click a certain point. User can use the selection box choose the type of unit and the number of unit. And then the information of this unit will be show in the information area.

Figure 8, Open txt dialog
5.3 Move reference unit

The third button in the status bar is “move reference unit” button. After clicking this button, it will show a dialog as figure 10. User can input the number of reference unit he/she wants to move. And user need to input the new coordinate of reference unit. The program will show the reference unit in new position and change the coordinate in file to new coordinate.
5.4 Show error

The fourth button in the menu bar is “show error” button. If user click this button, the program will pop up a dialog for showing error between real mobile unit and estimated mobile unit. In this dialog, you can choose three ways to show the error. The first way is using the graph to show the error between the real mobile unit and estimated mobile unit as showed in figure 11. The blue graph show the distance in horizontal direction from the real mobile unit to estimated mobile unit. The black graph show the distance in vertical direction from the real mobile unit to estimated mobile unit. So user can look up error of different unit through these two curves. The second way is using the table to show the error between real mobile units and estimated mobile units as figure 12. The first column in this table is the number of mobile unit. The second column is the error. This table can tell the user all errors of mobile units. The third way is using pie and table to show the distribution of error as figure 13. The first column is the range of error. The second column is the number of error located in corresponding range. The third column is the percentage of number in corresponding range. The right side of this dialog is the pie. It can tell the distribution of error to user.

![Figure 11, Show error dialog](image-url)
5.5 Select symbols and colors

At the beginning, program will use the default symbols and color to represent reference unit, real mobile unit and estimated mobile unit. In order to improve the flexibility, user can choose the symbols and colors they like to represent reference unit, real mobile unit and estimated mobile unit. Select symbols and
colors component will provide a dialog as figure 14 to allow user choose the symbol they like. User can choose the symbol and color they like to represent reference unit in the first row at the dialog and choose symbols and colors in the second, three row to represent real mobile unit and estimated unit.

![Figure 14, Select symbols and colors dialog](image-url)
6 Conclusions / Discussion

The project is to present visualization work of position system and develop each function module into component. This goal has been achieved successfully. The first goal is designing a simple and practical main UI which can meet all need of user. As showed in chapter 5, This main UI can show the map, reference units, real mobile units, estimated mobile units, legend of the map, information of units in a suitable way. This is desired result. The second goal is designing and developing each component of this project. Program divide all function modules into 5 components, map component, input parameter component, select symbols and colors component, move unit component, show error component, show information component. Each component is able to complete their respective function and work normally.

The third goal is visualizing the data of position system. As figure 9, all the reference units, real mobile units and estimated mobile units are showed into given position. And the legend and plotting scale are draw exactly. User can distinguish reference unit, real mobile unit, estimated mobile unit by the shape and color of symbols. Also, user can know the relation between real mobile unit and estimated mobile unit through the number assigned with the mobile units.

This visualization tool is also flexible and extendable. User can choose the symbols and colors they like to represent reference unit, real mobile unit, estimated mobile unit. There are two way for user to look up information of unit. User can choose the way they like. Also it provides four ways to show error between estimated mobile units and real mobile units. This program is also extendable. User can add the function they like by add some component. This project’s component also can be used in other project.

6.1 Future work

Although I have complete the goal of visualize the data from position system, there still are some drawbacks of my program. According to these drawbacks, I put forward the future work. This program should provide more flexibility. Now, this program just allow user choose different ways to show information. It is better to allow user design those ways. For example, when showing error between estimated mobile unit and real mobile unit, user can choose table to show the error, but user can not design this table. It is better that visualization tool should allow users design this table by themselves and show the content they want to look up. It should provide more components to user, so user can add function they like by adding these components to this program. For example, If user want to use three dimensional graphics to show the data. User can achieve this idea by add the component to this program.
6.2 Ethical consideration

This visualization tool can solve the visualization problem of indoor positioning system. This tool also open a door for developers. I can provide more interfaces to attract more people make contributions to indoor positioning system based on my work. Thus, it can solve visualization problem in a better way. I don't think this visualization tool will bring bad influence to people.
References

below is an example of an automatically numbered list of references according to the numbered list and cross references method, as described in chapters 2.4:

http://en.wikipedia.org/wiki/Qt_(software)

http://en.wikipedia.org/wiki/Visualization_(computer_graphics)#Scientific_visualization

http://en.wikipedia.org/wiki/Visualization_(computer_graphics)#Scientific_visualization


http://en.wikipedia.org/wiki/Indoor_positioning_system#cite_note-1

Visualization tool used in indoor positioning system
Huan Huang

2015-06-05