

**INTELLIGENT WALKING TOUR SYSTEM:
A System that Enhances the Services for Mobile Tourism**

by

Samrat Shakti Maskey
m

A Research Submitted in partial fulfillment of the requirements for the
degree of Master of Engineering in
Computer Engineering

A.I.T. LIBRARY

Examination Committee: Dr. Sumanta Guha (Chairperson)
Dr. Mathew N. Dailey
Dr. Nitin Kumar Tripathi

Nationality: Nepali
Previous Degree: Bachelor of Engineering in Computer Engineering
Nepal College of Information Technology
Kathmandu, Nepal

Scholarship Donor: AIT Fellowship

Asian Institute of Technology
School of Engineering and Technology
Thailand
December 2008

RSPR CS-08-05

ACKNOWLEDGMENT

I would like to tender my deepest appreciation and gratitude, first and foremost to Dr. Sumantha Guha, my research study adviser (chairperson) for his valued guidance, encouragement and support in the most difficult of times. My sincerest thanks are also extended to Dr. Mathew N. Dailey, and Dr. Nitin Kumar Tripathi, members of the Committee for their esteemed advice and understanding.

Special thanks to my friends, Miss Han Su Thet, and Mr. Sithisakdi Apichatthanapath. Thanks for the encouragement to finish my research and for defining the true meaning of friendship.

I would also like to thank Mr. San Lwin Htwe for his technical help and data.

To all my AIT friends, classmates, and people who helped me during my stay in AIT, thank you for cheering me up all the time and giving me wonderful life in AIT.

Finally, I offer my deepest respect and gratitude to my parent and whole family for their patience, unconditional love, and support. Without their encouragement, trust and understanding, I would not have had the endurance to complete this Masters Programme.

Abstract

Mobile tourism service is already contributing a lot in the development of tourism industry around the world. It is already providing location based services that could help travelers go around different places easily. Looking at all of these services and targeting a specific group of tourists, particularly, pedestrian tourists, we could add some more interesting service that could make their travel more comfortable, convenient and efficient.

The basic services that are typically provided at this moment are the location information of the user and routing information about how to travel to the next destination. This information is already making their travel easy and efficient, but we can add one more interesting service: personalized visual information that could really boost things up. Visual confirmation will with no doubt remove any confusion of the traveler's location. With this information, the user won't have to guess whether she is at the right location or not, she will be able to see it for herself. This knowledge of whereabouts can give a feeling of safety and the user can get the most out of their tour. This will really add the tour to be a memorable one.

Keywords: Mobile Tourism, Mobile Commerce, Mobile Information Service, Location Based Service, Context Awareness, Geographic Information System.

Table of Contents

CHAPTER TITLE	PAGE
Title Page	i
Acknowledgements	ii
Abstract	iii
Table of Contents	iv
List of Acronyms	v
List of Figures	vi
1. Introduction	1
1.1 Overview	1
1.2 Problem Statement	3
1.3 Objective	3
1.4 Scope and Limitation	4
2. Literature Review	5
2.1 M-Commerce	5
2.2 Global Positioning System	8
2.3 Geographic Information System	15
2.4 Mobile Application Development Environments	17
3. Methodology	18
3.1 Different Implementation Approaches	18
3.2 Determining Latitude and Longitude	21
3.3 Getting Information of the Location	22
3.4 Finding Shortest Distance and Direction between Two Points	25
4. Results and Discussion	27
4.1 Getting Location Position	27
4.2 Getting Information of the Location	29
4.3 Showing Globally Referenced Local Position	32
5. Conclusion and Recommendations	34
5.1 Conclusion	34
5.2 Recommendations and Future Work	35
6. References	36
7. Appendixes	39

List of Acronyms

API	- Application Programming Interface
ATM	- Automated Teller Machine
CDMA	- Code Division Multi Access
DGPS	- Differential Global Positioning System
DOD	- Department of Defense
DOP	- Dilution of Precision
GIS	- Geographic Information System
GNU LGPL	- General Public License Lesser General Public License
GPS	- Global Positioning System
JVM	- Java Virtual Machine
LBS	- Location Based Service
NMEA	- National Marine Electronics Association
OS	- Operating System
PC	- Personal Computer
PDA	- Personal Digital Assistant
SDK	- Software Development Kit
WGS84	- World Geodetic System

List of Figures

FIGURE	TITLE	PAGE
2.1	(General categorization of M-Commerce Applications)	6
2.2	(Satellites pin-pointing the location)	8
2.3	(Perimeter as location reference)	9
2.4	(Point of intersection as location reference)	9
2.5	(Point of intersection as location reference)	9
2.6	(Circle as the intersection of sphere)	9
2.7	(Two points as the intersection of sphere and circle)	10
2.8	(One point intersecting at the sea level)	10
2.9	(Multipath)	11
2.10	(Good Geometry)	12
2.11	(Poor Geometry)	12
2.12	(How DGPS Works?)	12
2.13	(GIS in action)	15
3.1	(Implementing Standalone Approach)	18
3.2	(Implementing Web Based Approach)	19
3.3	(Implementing Hybrid Approach)	20
3.4	(Internal database structure)	22
3.5	(Getting Geo-Tagged Images)	23
3.6	(Different data layers)	24
4.1	(Starting the application)	27
4.2	(Getting location)	27
4.3	(Stopping the GPS)	28
4.4	(Re-start GPS)	28
4.5	(Destination Menu selection)	29
4.6	(Destination image information)	29
4.7	(Enlarged destination picture)	30
4.8	(Destination with description)	30
4.9	(Nearby Images)	30
4.10	(Accessing data from internet)	31
4.11	(Nearby geo-tagged Flickr images)	31
4.12	(Globally referenced local position)	32
4.13	(Zooming feature)	32

4.14	(Zoomed in map)	32
4.15	(Loading the travel route)	33
4.16	(Route loaded as green color line)	33

Chapter 1 Introduction

1.1 Overview

Following the developments in the field of mobile internet and device technologies, there has been a rapid increase in the use of internet-ready hand-held GPS-enabled mobile phones, smart phones and PDAs. This has triggered a rapid growth in the development of applications related to them [1, 2]. Applications like context aware systems, financial systems, entertainment systems, and health systems are some of the basic and common applications that are becoming popular in these kinds of system [3]. Finding the location of your contact, paying money for the transaction, watching the trailers of a movie or updating your health record could now be done from anywhere with a single press of your mobile device. This gives us the power of location independent availability. And with this power, we can still do a lot to raise the standard of our living. All of these applications and innovations are shaping up the future of mankind and how we will communicate and do business.

Most of these applications are advancements in the existing technology of e-commerce. With the feature of ubiquity (i.e. being available at anytime anywhere.), and transaction being conducted over the mobile terminal, we call this type of system m-commerce. And all of the above mentioned applications can be classified under the m-commerce technology. The value added attributes of m-commerce like ubiquity, convenience, intractability, flexibility, personalization and localization are ensuring the growth and development of these kinds of applications [3]. These attributes not only help in the advancement of the technology but also provide exceptional potential of growth, productivity and profitability in a business world. No-wonder why a lot of people predict that m-commerce will be one of the dominating forces for business and society in near future.

In addition to the value added attributes of m-commerce, the development of these kinds of applications are driven by factors like: wide range of availability of mobile network, large number of users of mobile devices, cell phone culture among youths, demands from service oriented customers, location based services, vendor marketing, declining price of service, improved performance and bandwidth [3].

One of the application that is becoming more and more popular day by day as the technology of high sensitivity GPS advances is context aware system. Applications like search navigators and mapping systems are some of its examples. Searching which is the nearest restaurant or which is a recommended restaurant falls under search navigators and tourist guide systems can be an example of mapping system. These applications are the result of Location Based Service (LBS). This type of services will first identify the location of the person and then provide relevant information about that place. It is really helpful in real-life situation and with the advancement in GPS technology; we can expect the precision of the calculated location to be around five meters or less. So, we can imagine why the popularity of this kind of applications is increasing.

Talking more on this kind of applications, there are already existing systems which will navigate you to certain destination or help you visualize the locations. There are also some map applications which will help different kind of users to navigate around the specific

location. People who are new to that place could have a huge benefit from this kind of application. Focusing on a specific group of users among these people, tourists can use this kind of applications to make their travel easy. And with the increase in the number of pedestrian tourists on the road we can focus on them and develop an application that could help in making their tour comfortable, convenient, and efficient. We can implement the above technologies and integrate the location information of the user with the information about the place such that the user can get the most out of their tour. We will call this application "Intelligent Walking Tour System".

In order to implement this system, we would need precious location information of the user. We could get this information using the cell based triangulation or using the GPS. Both techniques use the triangulation method to calculate the location. We would also need the information about that location. To get this information we would use Geographic Information System which will tell us what is located at that place. And finally a mechanism, that will efficiently integrate these two information. Since we are implementing this system in a mobile network, so we also need to know the basics of M-Commerce

In this report, we will learn the basics of systems required to implement this application. Understand why we need them and how they work. And finally implement a system as a proof of concept showing the importance of destination information in terms of image. In this process, we will try to obtain the latitude and longitude data from a GPS receiver and then plot them on a map. This will pave our way towards implement this application "Intelligent Walking Tour System".

1.1.1 Intelligent walking tour system

Sometime when we travel in a new or unknown place, then we would need some kind of guide to help us go around that place. If not, then we may feel a little unsafe and unsure about whereabouts of our location. Looking at those tourist maps and not knowing where we are, makes us more frustrated. Those signs in unknown languages and local peoples who don't understand us make us more confused of our location and where to go next. We may feel that we are lost and try to return to the place we already know. This is what happens to the general tourists who just want to explore a new area themselves.

But, this is not what we wanted from our travel, we came to travel not just to go around the known places, but to discover the great unknowns, may be the place, culture, architecture, etc. Just because we are unsure of whereabouts of our location and also of the new location, we cannot give up our dreams. One way to continue with our dream is to hire a guide and just follow him. Another way, which is more interesting, cheap, safe and more fun, is to use Intelligent Walking Tour System.

Telling us about the exact location of where we are and where to go next, how to go there, what to expect to see there at different time, or just to visually confirm the place where it says you are, is really that place or not by showing the pictures near you, and guiding you how to go to next place. This is what Intelligent Walking Tour System does, and this is what a general tourist would need to explore in a new place. A feeling of safety and the knowledge of whereabouts really adds to being a memorable travel.

In this age of mobility and technology, people are generally equipped with the power of technology that can help them do a lot of thing anywhere in the earth. One of the examples is mobile phones, which are generally equipped with a GPS receiver. Now people only need some intermediary software that will translate those technical details into human readable form. One of these intermediaries is "Intelligent Walking Tour System". User friendly, and very easy to use, this system is developed specially for tourists who want to explore around an unknown territory by themselves, or just walk around the famous places.

1.1.2 Related Commercial Services

There are a lot of commercial products and applications that can show the location information along with different ways to navigate the area. Applications like SpeedNavi, SatNav, GPSPilot, PDAFun, TomTom, Gramin, and a lot others which if I try to mention will fill many pages are already providing services that could facilitate the travel and give the location information along with different other features like searching nearby locations, voice guidance, etc. Not only this, but there are other features like intelligent search and agent based search being developed so that it will provide the user with the better and user preferred results.

1.2 Problem Statement

Mobile tourism is already providing some of the basic services like the location information of the user and rout information about how to travel to the next destination. This information is already making the tour easy and efficient. But even with this information, sometimes the user cannot be certain about his exact location. He has to ask someone to confirm whether he is where he is supposed to be.

An adaptation in the new GPS chipsets for use in urban areas, the GPS receiver tries to reduce the error from multipath effect by freezing the location information of the user and try to get a better fix if they are not moving above certain speed (3-4 km/hr). This type of feature is called static navigation and is specifically seen in some GPS chips manufactured for urban areas. This will certainly give better results in terms of accuracy if we are static and not moving or moving in high speed, but this is the worst case if we are using it for something like walking tour system. We either have to switch off this feature and tradeoff with the accuracy of the location or start to walk above the speed limit to keep the dynamic location information.

1.3 Objective

We can add interesting service like the visual information to the already existing mobile tourism service. The destination information in terms of personalized visual confirmation, will with no doubt remove any confusion of the traveler's location. With this information, he won't have to ask anyone about the location, he will be able to see it for himself.

The specific objectives of this research are to

1. Provide the destination information in terms of description and images along with the time specific information.
2. Study various approach to deliver the destination information.
3. Show globally referenced local position on a map along with the surrounding information and direction.

1.4 Scope and Limitation

The limitations and scope of this study are as follows:

1. While developing any location based service, there are a lot of ethical, legal and health related issues which is beyond the scope of this research.
2. One limitation of this project is that it is target to a small group of users and even among these users; we expect them to have some kind of internet ready hand held GPS enabled device. This is too much to expect from pedestrian tourists.
3. We scope down the application to run in only one platform at this time because the application will be device dependent.
4. While displaying the destination information in form of geo-tagged images, a lot of non-relevant images may also show up.

Chapter 2 Literature Review

2.1 M-Commerce

"The core of mobile e-commerce is the use of a terminal (telephone, PDA, PC device, or custom terminal) and public mobile network (necessary but not sufficient) to access information and conduct transactions that result in the transfer of value in exchange for information, services or goods." -Ovum [4]

The widespread availability of the wireless and mobile networks, mobile devices and software that make these devices more usable and secure is creating a new window of opportunity to perform different types of transactions that could have only been done in a desktop computer. With an access to real-time information, we can perform all these transactions through a mobile device. This technique of performing any kind of business activity or transaction over a mobile or wireless network through a mobile device can be referred to as m-commerce. [2-4]

The driving forces behind the rapid development of this field are some of its basic attributes and characteristic. These attributes are contributing in creation of applications that are only possible in mobile devices [3]. Some of them are as follows:

- Ubiquity
- Convenience
- Intractability
- Personalization
- Localization

Ubiquity is being available anywhere at any time. This means mobile devices can easily access and deliver information anytime and anywhere making it a fastest way of information communication in real world. This gives a lot of advantage in many fields.

It is easy and convenient to work in a mobile environment. Since it is very easy to connect to the internet, intranet or other mobile devices, it is very easy to access different types of information. So, mobile devices could become one of the most convenient and preferred medium to access the information.

Transaction, communication and service provision in a mobile network is very fast and interactive when compared with other network. With this ability, applications related to service and delivery will highly benefit from this environment.

Mobile devices are generally owned and operated by single user. With this personalization, information and service that meets the requirement of the individual can be delivered easily and efficiently. Like amazon.com uses the personalized information to recommend the books to the users, there could be millions of such applications but only more accurate and to the right person.

In a mobile network, we can know the exact location of the person. With this information, we can offer relevant services to them at that time. This type of service is called l-commerce. A lot of new applications could be thought in l-commerce, which is only possible in mobile devices.

With these features it is no wonder why people are talking about the dominance of mobile application in the business world and overall growth and development of the human being.

2.1.1 Services in m-commerce

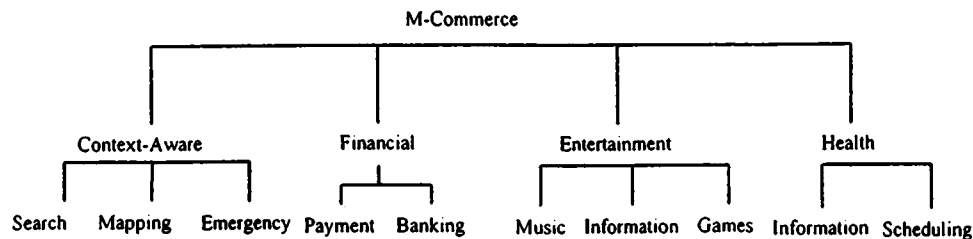


Figure 2.1 (General categorization of M-Commerce Applications)

If we think of services that could be provided based on the attributes of m-commerce, then there are a lot of them. So we try to identify the possible applications and classify them as much as we can. [3, 5]

Thinking of it, the first type of applications that would come in our mind would be context aware systems. Using the GPS technology or similar technology like triangulation of radio or cell based stations; we can get the exact location of the users. Then we can deliver the service, information or product to the customer's location. Both the customers and businesses benefit from this type of application. Location information can be used in applications like emergency services where the exact location of the customer is required. It can also be used in navigation where we can locate what is nearby. Also we can track the movement of a person or thing or use it in mapping like tour guides. [3, 6, 7]

As we are planning to develop a tourism service, we will dig a little more into the service for mobile tourism. This service is becoming popular and essential day by day as we use it and as the mobile technology improves. Not just for a new tourist but even for locals who want to visit a building which he is not sure of the exact location; this could be really helpful. They can use this type of service to travel to that place very easily. Some of the features which could be helpful in this service like Map, Information on Destination, Tourist Attraction Information, Transportation Information, Emergency Contact, etc. should be included where as the features that may not prove to be as helpful like Gaming Activities, Digital Souvenirs, Travel Stories, Speech to Speech Translation, etc. should be avoided. [1, 6, 8, 9]

Another important area where mobile applications can boom is financial area. Soon, the bank branches, ATMs, credit cards could be replaced by mobile devices. Applications like mobile banking let the customers to use their mobile to access the financial information, pay bills and transfer funds. Applications like m-wallet, Nokia-wallet, etc lets users make purchase with their mobile device and store information like visiting card and credit card numbers. [3]

Entertainment is a huge industry in itself. Mobile entertainment is another area where applications related to music, video, games gambling and more are emerging and are doing

good. Other than that, information related to entertainment can be easily delivered to users based on the personalization and localization. [3]

Health is another field where m-commerce can create an impact. Applications that can help and assist in dispatch function of emergency call or emergency response cell phone calls are some of the examples where the usability of mobile devices can be used. Other applications like information of health of a person can be stored in the device which could be used by the doctors when needed. [3]

Off-course, there are a lot of ethical, legal and health related issues related to m-commerce which will have to be resolved as we head towards this world of mobility. But we will not be discussing about them here. Rather we will try to imagine the mobile world and how efficient it would be if we can implement the applications we just discussed.

2.2 Global Positioning System

2.2.1 What is GPS?

GPS is a system that determines the exact location of a suitably equipped object at that specified time. It is actually a constellation of twenty-seven earth orbiting satellites (twenty-four in operation and three extra in case one fails) at an altitude of approximately eleven-thousand miles above the earth surface transmit signals that allows GPS receiver to calculate the location of their position through trilateration. When we talk about GPS, it generally means a GPS receiver. It is an electronic device which listens to signals from the satellite and uses the information to determine and display the receiver's location, speed, altitude and heading. Basically, GPS receiver calculates how far it is from at-least three overhead satellites and then does some geometric calculation to find its two-dimensional position. [10-15]

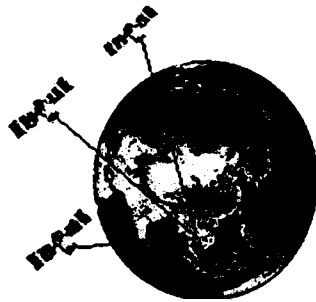


Figure 2.2 (Satellites pin-pointing the location)

Officially named as NAVSTAR GPS, this fully functional Global Navigational Satellite System was developed and is operated by the United States Department of Defense (DOD). Initially intended only for military applications, the government made this system available for civilian use in 1980. Since then, GPS has been widely used in navigation worldwide, and is a useful tool for travelling, mapping, surveying, location-based commerce (e-commerce), scientific uses, and other applications where precise time reference relating to that position is necessary. [11, 16]

As the GPS units are becoming smaller and cheaper, at the same time different techniques and technologies are also improving the performance of the GPS device. A simple uncorrected position determined by the GPS receiver produces accuracy in the range of fifty to hundred meters, but using the technique called differential correction, we can expect the accuracy of the device in the range of five meters or below. Even more, using the advanced hardware technologies in GPS, we can expect a longer battery runtime and more sensitivity so that we can even use the devices in the urban areas with tall buildings. [17]

We can say that the true utility of GPS is when it is combined with GIS. Basically GPS tells us where we are and GIS tells us what is in that place. So GIS is like a descriptive database of the earth and together with GPS is changing how we look at the world.

2.2.2 How does it work?

Suppose, we know the distance between us and a fixed place, then that means we can be anywhere in the perimeter of that circle with radius equivalent to the distance and center as the fixed place as shown in *Figure 2.3*.

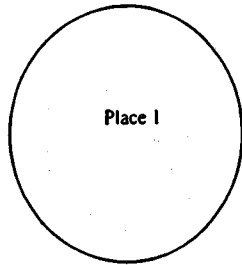


Figure 2.3 (*Perimeter as location reference*)

Now suppose we found out the distance from another fixed point, then we can create another similar circle with new radius and centre. When we combine this information, then we have two circles that intersect at most two places. So we must be in one of the intersection as shown in *Figure 2.4*.

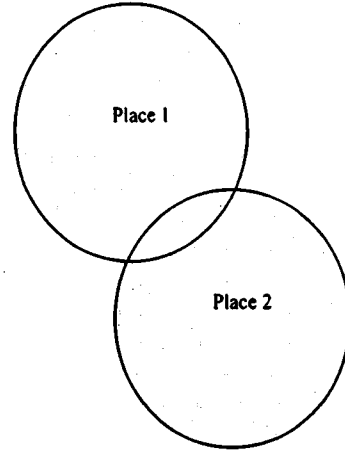


Figure 2.4 (*Point of intersection as location reference*)

Now if we found the distance between us and a third fixed point, and then similarly creates another circle and combine the information, then all the three circles will surely intersect in a single point eliminating the doubt of where we are. And this point of intersection is shown in *Figure 2.5*.

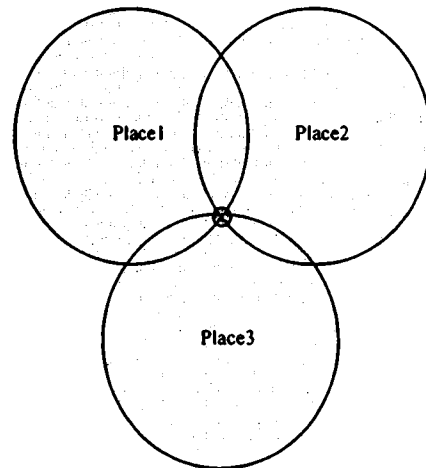


Figure 2.5 (*Point of intersection as location reference*)

Similarly, if we consider sphere instead of circles we would be able to calculate the exact location in the real world situation. Here, when we intersect two sphere then the points of intersection will create a circle where we can be anywhere in that circle. This can be seen in *Figure 2.6*. Now when we intersect the third sphere, then it will intersect the created circle in at-most two points. This can be seen in *Figure 2.7*. Now if we consider earth itself as the fourth sphere, only one of the two points will be in the surface of the earth as seen in *Figure 2.8*. So we will be able to

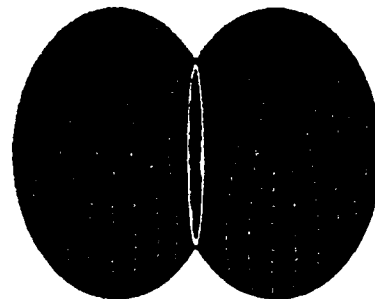


Figure 2.6 (*Circle as the intersection of sphere*)

determine the correct location of a GPS receiver with only three satellites. Now if we need more precious information regarding the altitude, then we would need at-least four or more satellites. [10, 12, 13, 16]

To do this calculation, the GPS receiver must know the exact location of the satellites above it and the distance between it and each of the satellites it has a fix. The satellites transmits some radio signals which is located by the GPS receiver and upon receiving these signals it calculates the distance using the travel time of the radio signal, and also performs the CDMA (Code Division Multi Access) to identify the individual codes. In this way, the GPS receiver is able to determine the precious location and data of each satellite.

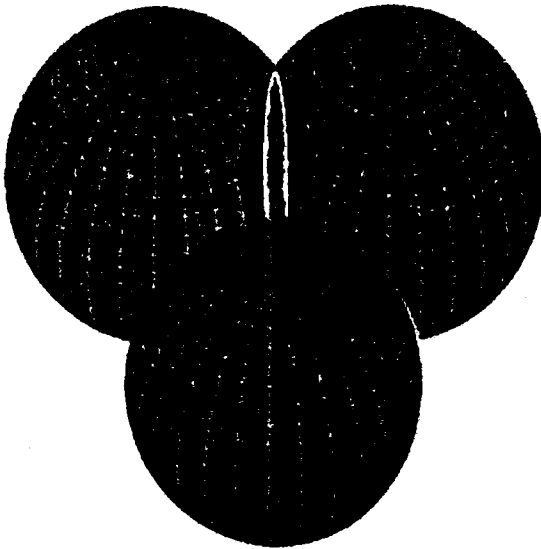


Figure 2.7 (*Two points as the intersection of sphere and circle*)

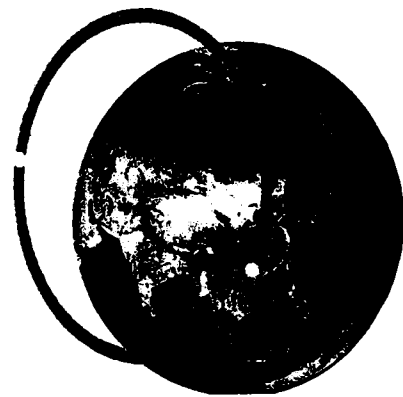


Figure 2.8 (*One point intersecting at the sea level*)

2.2.3 How it calculates the distance?

If we calculate distance in terms of speed, then it is the product of the speed per unit time. And if the speed is constant then it is the product of speed and total time to reach the destination.

$$\text{Distance} = \text{Speed} * \text{Time}$$

GPS uses this simple formula to determine the distance between the satellite and the receiver. These GPS signals are the radio signals which have the speed of light which is about 186,000 miles per second. The GPS receiver measures the amount of travel time of the GPS signal from the satellite to the receiver. Then it can calculate the distance between it and the satellite.

In order to do this, the receiver and the satellite both needs to be synchronized to nanoseconds. But the receiver does not have this much accurate clock, so it looks at the incoming signal which has the information about the atomic time and orbit position. After calculating at-least four different signals, it resets its clock to the atomic time of the satellite; it can finally get the nanosecond precision. Then it calculates the distance. [10, 12, 14, 16]

2.2.4 Source of error

- Selective Availability

It is the intentional alteration of the time and ephemeris (orbit position) signal by the Department of Defense. It generally introduced about seventy meters of errors. But fortunately, it was deactivated since May 2, 2000 due to the broad distribution and worldwide use of GPS. Anyway, using DGPS helps reduce this error to about five meters.

- Atmospheric Effects

The speed of GPS signal in the space and in the atmosphere is not the same. The ionosphere and the troposphere refract the radio signal causing it to travel at a different speed. This causes an error in the calculation of distance from "speed*time".

- Multipath

When receiving signals in an urban area or dense area, then the signals may reflect with large objects and confuse the receiver about the exact time and location. This is shown in *Figure 2.9*.

- Clock Inaccuracy

Although we sync the receiver clock with satellite, there are still some inaccuracies in time that may cause some error.

- Satellite Orbital Position

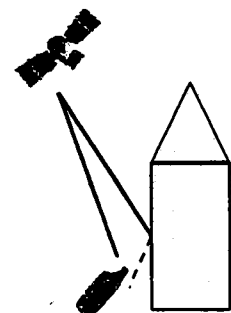


Figure 2.9(Multipath)

Although the satellites are moving in a fixed orbit, slight shift in these orbit are possible due to the gravitational force of earth, moon and sun.

- Satellite Geometry (Dilution Of Precision)

Different satellite geometry can increase or decrease the accuracy of the location. Greater angles between the satellites lower the DOP and increase the accuracy. Small angle between the satellites increases the DOP and reduces the accuracy.

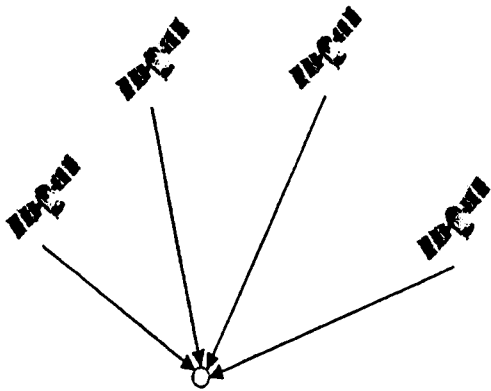


Figure 2.10 (Good Geometry)

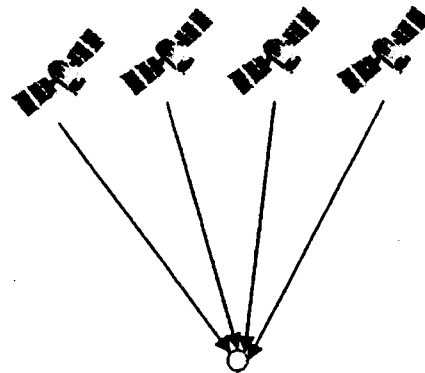


Figure 2.11 (Poor Geometry)

2.2.5 Differential GPS

A normal GPS works by finding the distance between at least three satellites and then triangulating with these satellites. When finding this distance, there could be a lot of error due to the variation of travel time of the radio signal in various medium and so on. So if we know our exact location, then we can calculate these errors. This location whose exact position is known is made a ground station and it broadcasts the error correction factor for each satellite which it used to calculate the new location data. Figure 2.12 shows how a DGPS system works.

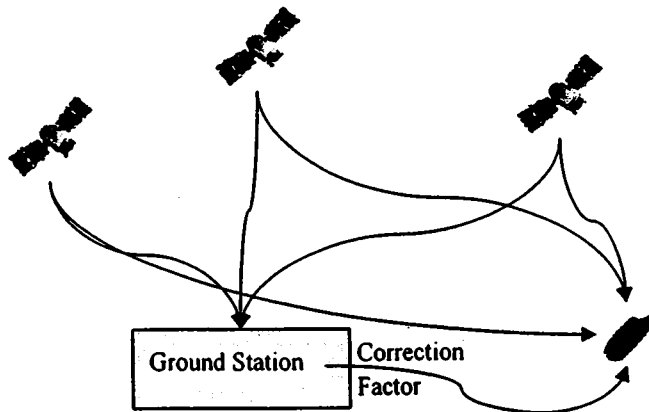


Figure 2.12 (How DGPS Works?)

Here first the precious location information of the Ground station is already known. Then it recalculates its location and checks the difference with its known location to calculate the correction factor. Then it broadcast this factor to all the DGPS receivers which will apply this factor while calculating its location.

The basic principle of operation of DGPS is that when two GPS which are near to each other then they will generally face similar atmospheric errors. And when the location information of one of the GPS is known, then it can calculate the error and then apply this error factor on to the other GPS to reduce the error on the other GPS [12, 14].

So to achieve an accuracy level below ten meters, we use differential correction factor. We generally use a real-time differential correction but it could vary according to project. We could also use post-processing of the collected data. So, implementing it would completely depend on the type of project and the availability of the data for correction factor.

2.2.6 NMEA protocol

"NMEA 0183 (or NMEA for short) is a combined electrical and data specification for communication between marine electronic devices such as echo sounder, sonars, anemometer (wind speed and direction), gyrocompass, autopilot, GPS receivers and many other types of instruments. It has been defined by, and is controlled by, the U.S.-based National Marine Electronics Association." [18]

In general, we use NMEA sentence defined by the NMEA protocol to communicate between the GPS receiver and the satellites. This sentence contains all the information requested by the GPS. There are different types of sentences depending on the need of data. So we should be careful when choosing the sentence. A generalized version of the sentence would look like this:

`$aabb,d1,d2,.....,<CF><LF>`

Here \$ specifies the beginning of the sentence. The first two letters "aa" will identify the talker (i.e. GP=GPS, AP=Auto Pilot, etc). Then the next three letters "bbb" identify the sentence (i.e. GLL=Geographic Position Latitude/Longitude etc). Then different data according to the sentence requirement follows which are separated with commas. At the end of the sentence, there is generally a checksum and carriage return. [19, 20]

In addition to these standard sentences, NMEA also allows different companies to define their own sentence which only their devices can use. Generally, this type of sentence begins with P and then the three letters identify the company. For example, a Garmin sentence will begin with PGRM.

Some of the sentences that may find its applicability with GPS are shown below: [19, 20]

- GPAAM - Waypoint Arrival Alarm
- GPALM - Almanac data
- GPAPA - Auto Pilot sentence "A"
- GPAPB - Auto Pilot sentence "B"
- GPBOD - Bearing Waypoint to Waypoint
- GPBWC - Bearing any Distance to Waypoint – lat/long, N/S, E/W, UTC, Status
- GPDTM - Datum Reference
- GPGBS – GPS Satellite Fault Detection
- GPGGA – GPS Fix Data
- GPGLC – Loran-C data
- GPGLL - Lat/Lon data
- GPGRS - GPS Range Residuals
- GPGSA - Overall Satellite data
- GPGST - GPS Pseudo range Noise Statistics
- GPGSV – Satellites in View
- GPGTD – Geographic Location in Time Difference
- GPMSK - send control for a beacon receiver (for DGPS Beacon Receiver)

GPMS - MSK Receiver Signal status
GPRMA - Recommended Minimum Navigation Information
GPRMB - Recommended Minimum Navigation Information
GPRMC - Recommended Minimum Navigation Information
GPRTE - Routes
GPTLL - Target Lat/Long
GPTRF - Transit Fix Data
GPSTN - Multiple Data ID
GPVBW - dual Ground/Water Speed
GPVTG - Track Made Good and Ground Speed
GPWCV - Waypoint closure velocity (Velocity Made Good)
GPWNC - Distance - Waypoint to Waypoint
GPWPL - Waypoint Location
GPXDR - Cross Track Error - Dead Reckoning
GPXTE - Cross Track Error - Measured
GPZTG - Zulu (UTC) time and Time to Destination Waypoint
GPZDA - Date and Time

So, when we choose a sentence, we should be careful not to choose a sentence with extra data. For example, if we just need latitude and longitude information then we could just use \$GPGLL and not other sentence with a lot of extra data along with latitude and longitude.

2.3 Geographic Information System

A Geographic Information System is a type of information system which tells us about the details of what is located at a place. But just knowing the location information is not much of a value when we don't know where that place is. So when we link this information with the location details we extracted earlier, a lot can be done with this knowledge. *Figure 2.13* illustrates this concept. [11]

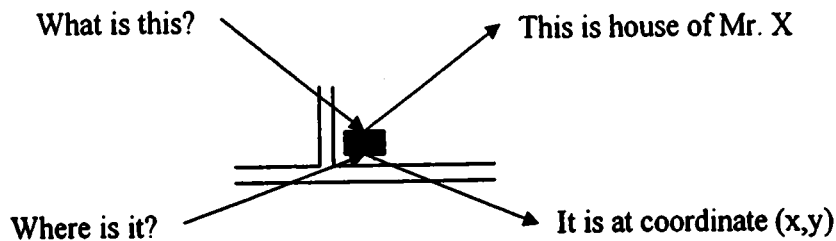


Figure 2.13(GIS in action)

This location information which tells us about the details of the place is generally referred to as the features of that place. For example a house or a statue can be feature of a place. These features generally have some attributes and values. For example, the attribute of the house could be its color, cost, size, etc and its respective value could be white, \$300,000, one story building with 2 bedrooms, etc. And when we collect the similar attributes of the features then a huge database is created which can be categorized based on the attribute values. For example we can categorize a house based on its cost or size. And when we link this information with the location details, then it gives us the base of the Geographic Information System.

When we try to link these features with the location details, we can do it in three ways:

- Point
- Line
- Area or Polygon

When we try to link the feature according to point then this feature is generally linked with a single GPS position. So geographic place with attributes and characteristics but not sufficient area to be displayed on a map are displayed using the point and will have a single point of reference or GPS coordinate. For example a house could be displayed as a point.

When we have to link similar features then it is generally linked with a collection of GPS position forming a line. A geographic place made of connection of similar attributes and characters but not sufficient width to be displayed on a map are displayed using the line and will have connection of GPS coordinates. A road or river could be displayed as a line in a map. Generally some spatial locations could be on either side of the line. For example, a house could be beside a road. [11]

When we have to link similar features with some distinct area then it is generally linked with a collection of GPS positions and the beginning and the end of points being the same, forming a polygon. A geographic place made of connection of similar attributes and with

sufficient area to be displayed on a map is displayed using a polygon. So it is similar to line except that the end of the line are tied together to form a closed area. Generally area contains a collection of spatial locations and points. It could be either inside an area or outside. For example, AIT is outside Bangkok.

So after we have linked the features with the location details then we can do a lot of things like visualizing the data and knowing what would happen if something happens and use it for different purpose efficiently and easily.

2.3.1 Advantage of GIS

GIS is a rapidly growing technological field that incorporates graphical features with tabular data in order to assess real-world problems. It is used to create, store, analyze, retrieve, manipulate, and output geographically referenced data or geospatial data in order to improve the overall information infrastructure, which directly impacts the overall growth and development. [11]

There are a lot of advantages of GIS because of which it is widely spreading in other fields. One of the main advantages of GIS is that it helps to gather and integrate different data so that it can tell us what is exactly in that place where the GPS locates the point. This may help in making better decisions for planning and managing facilities, transportation and trip.

It also allows us to visually analyze and identify the location, leading to a better understanding of what we should expect. It helps us to make better decision. "Better information leads to better decisions". It can be used to assist in tasks such as presenting information.

So it is this power of GIS which has added so much value to GPS. Without it, only the location detail is not much of value. And Without this exact location detail, it is not so efficient to use what is in that area also.

2.4 Mobile Application Development Environments

As the mobile technology develops, a lot of new environments are evolving in this field. Developing an application would require exact specification of that environment. We can choose between BlackBerry, Windows Mobile, Palm OS, Symbian, Linux, J2ME, Android, and others. We need to choose the environment such that the application has the greatest applicability and usability among these communities. To do this we need to consider some specific attributes that will help us choose which platform to build application for. Attributes like application's performance, security, development time, number of users etc should be consider before anything [21]. Some of the approaches that could be used to develop mobile applications are:

- Using J2ME development platform
- Using platform's native language

J2ME is an ideal development platform for developing applications that are platform independent. It reduces the development effort to make the application ready for all possible platforms and devices but the devices must have JVM on them. Some of the drawbacks of this approach are the application performance will not be at its best and the security of the application may be compromised as it runs on top of JVM [21, 22].

We can also consider developing applications in the platforms native language. If we do this, then the first advantage we will have is the applications performance. Also we can improve the security of the system. Some of the drawbacks of this approach are the development effort and platform dependency. But there are already existing projects that are trying to remove the platform dependency in platforms native language. For e.g. Net60 is an implementation from Red Five Labs to bring .Net Compact Framework applications unchanged, to Symbian platform (the OS running the Nokia and Sony Ericsson Smartphone) [21, 23].

So it's up to the requirement of the user on choosing the approach of developing the mobile applications. In our case, initially we had a windows mobile device so we choose a platform specific language: .net compact framework. Although it is platform dependent, there is a project that is implementing it in Symbian platform without having to change the source code. This will reduce the development time of the application and at the same time increase the overall performance of the application.

Being more specific, we used c#, .net compact framework as the development platform along with windows mobile 6 SDK. We also use the shape file and Microsoft SQL Server 2005 Mobile Edition to store the data locally. Then we implemented this system and deploy it in ASUS P526 Windows Mobile 6 powered PDA where we successfully tested the system.

Chapter 3 Methodology

The main purpose of this application is to provide a specific group of user (Pedestrian Tourists) with the information they will need to travel around a place and the destination information. This will not only help them to travel around but also provide visual information such that the users will be able to visually confirm the place they are visiting. In order to implement this system, we will need precious location information of the user. We can get this information using the cell based triangulation or GPS. Since new mobile phones are generally equipped with GPS Receivers, so we will use this technique to pinpoint the location. We will also need the information about that location. To get this information we will use Geographic Information System, which will tell us what is located at that place. And finally a mechanism, that will integrate these technologies. But before discussing in details about them, we will look at different methodologies to implement it.

3.1 Different Implementation Approaches

When implementing this system, we can use various methods to integrate the information about that location. All of these approaches have their own advantages and disadvantages. We will look at various integration methodologies and find their advantages and disadvantage.

3.1.1 Standalone approach

As the name suggests, this approach will be able to operate without any help from outside services. When developing application with this approach, we will either have to create our own map or buy the map. After that, we have to calibrate it according to our requirement and link with the location information. This approach will store the entire database in a single location inside the device. The application will simply access the database and display the relevant information.

When implementing this approach, the applications performance will be fast, efficient, and secure. But some problems that could affect this approach may be platform dependency, difficulty in updating the information in already deployed devices, cost of developing the database and the memory required to store the database. We could rule out the memory criteria as memory is getting cheap and more available in new devices, but rest of the problems still remains.

3.1.2 Web based approach

As the name suggests, this approach will use the web service to integrate the data with the application. We don't have to develop the entire database ourselves. We can just use the existing map database provided by different map servers like Google map or Map server,

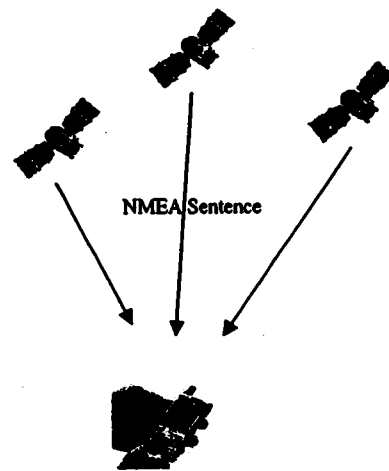


Figure 3.1(Implementing Standalone Approach)

etc. and integrate the location specific information database which is stored in some application server. This approach will get all the required data from the web server and display the required information as needed.

When implementing this approach, our data will be up-to-date and stored in server. We won't have to worry about the extra space to store the data, and creating a database will not cost as high as in standalone approach. But we do need to care about high speed internet as all our data and information will be transferred through the internet. As internet is sometimes described as hostile environment, so we also need to take care of the security of data and application.

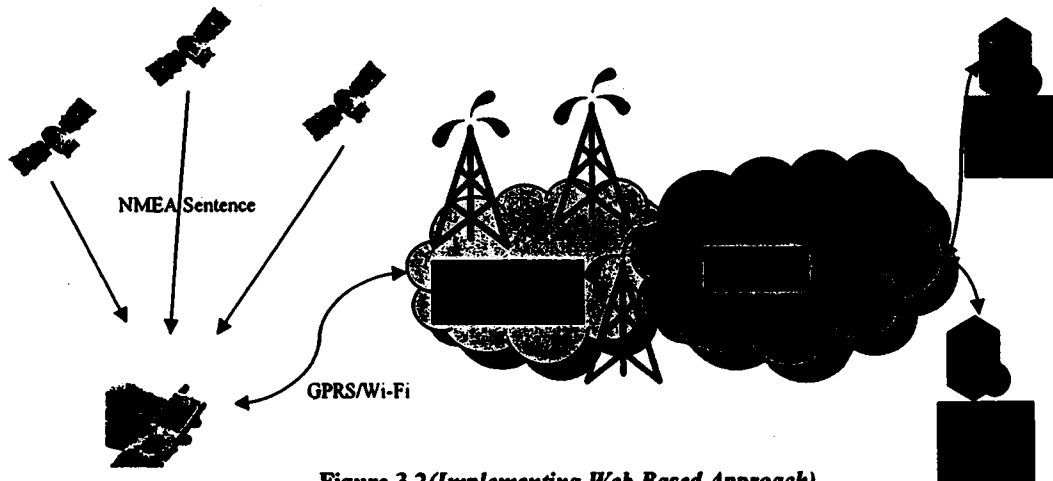


Figure 3.2(Implementing Web Based Approach)

3.1.3 Hybrid approach

As the name suggests, this approach is a hybrid of standalone and web based approach. The user can cache the location information of the place they are planning to visit when they synchronize the device with laptop or pc. As the pedestrian tourists won't be able to travel a large distance in a day, so we won't need a large storage to cache the data and in the same time the application performance will be good as the required data is already stored in the device. If in case, the user travels further, then he can always switch back to the web based approach to get the required data as needed.

When implementing this approach, the applications performance will be fast, and efficient. Also, we won't need large space to store the entire database as we will only cache the required data and extract the required data as required. We won't even need the fast internet connection until we are travelling in the region that is already cached in the device. But if we go out of the region, then we will need the internet connection to update the devices location information. Also our data will be up-to date.

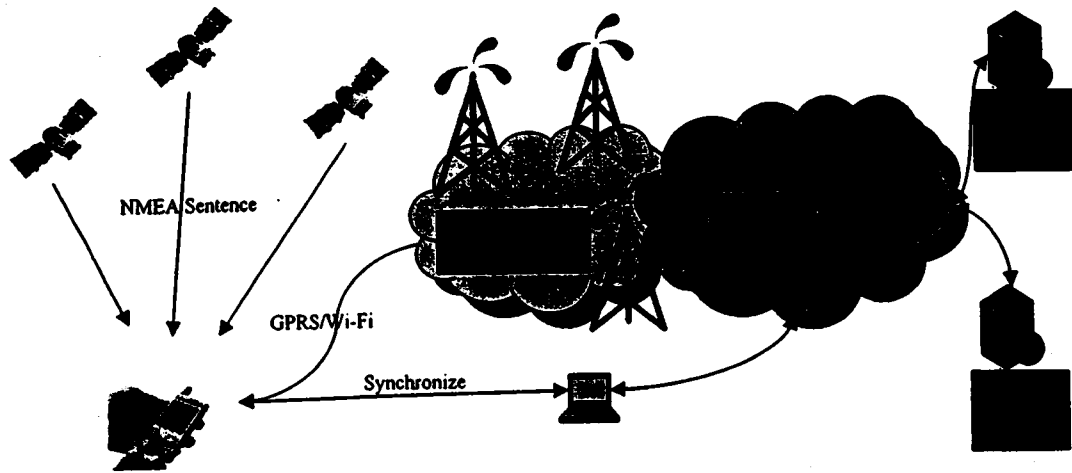


Figure 3.3 (Implementing Hybrid Approach)

3.1.4 Comparing the approaches

Table 3.1

	Stand Alone	Web Based	Hybrid
Internet Connection	Not Required	High Speed Connection	Not Required (depends on situation)
Performance	Excellent	Good	Excellent
Data Available	Installed Data	Up-To-Date Data	Up-To-Date Data
Storage Required	Large	Small	Small

When we look at standalone approach, the applications performance is excellent and it doesn't even need internet connection. But it requires large storage space and the data is not up-to-date. The web based approach eliminates these drawbacks but in the cost of reduced the applications performance and requiring high speed internet.

The hybrid approach tries to remove all the drawbacks of standalone and web based approach. It requires small storage but still the data is up-to date and the applications performance is excellent. In general case, it doesn't require the internet connection but in some special cases where there is not enough data available in the cache, then the user have to connect to the internet to get the required data. So it has to switch back to the web based approach.

As a proof of concept, we will try to implement a standalone application which will access the local database to show the destination information. This will be the base even for the hybrid approach which will access the data in the cached format. Then we will try to access the data in the web.

3.2 Determining Latitude and Longitude

The first thing I tried to do was to get the latitude and longitude data from the satellite. Doing this would have been a difficult task but because of the standards introduced by NMEA (National Marine Electronics Association) which is used by a vast majority of the GPS receivers, we just needed to interpret these NMEA sentence. A simple NMEA sentence would look like this

```
$GPGLL,1111.11,a,yyyyy.yy,a,hmmss.ss,A*hh
```

This sentence basically contains the entire data we need to finish our first task, i.e. get the location of an object. If I interpret this sentence, then \$ is the start of the sentence. GP is Talker Identifier for GPS. GLL is a Sentence Identifier for Geographic Position – Latitude/Longitude. We could have used other Sentence Identifiers like GGA, GLC, etc but they return a lot of data that may not be used at this time like speed, distance etc. and increase the traffic and time to interpret. So we try to use sentences with minimum length and useful data for the time. The comma separates the data. The first data is Latitude, the second is N or S (North or South), third is Longitude, fourth is E or W (East or West), fifth is Time (UTC), and the last data is for data validity through checksum. [19, 24]

Windows mobile 6 SDK provided an intermediate Driver to handle the GPS. This intermediate layer acts as a gateway to access the GPS receiver. It gives us ability to access the data from the GPS as if we were communicating with the actual GPS. Because of this many applications can use the GPS at the same time. This enables us to write codes that will work with any GPS hardware and vice versa. [25, 26]

And with the introduction of GPS intermediate Driver, I just use the parsed API to access the parsed data directly. The GPS intermediate driver's parsed API already parses the NMEA sentences into the individual data. We just used these APIs to retrieve those GPS data;

GPSOpenDevice, GPSGetPosition, GPSGetDeviceState, GPSCloseDevice

GPSOpenDevice handles back to the intermediate driver. Using this handle we can get continuous location information using GPSGetPosition. Also, this intermediate layer allows multiple applications to use GPS hardware at the same time. The GPS Intermediate Driver makes it appear that each application has its own dedicated GPS hardware and a single application (Intermediate driver) accessing the actual physical device. [26]

When we get the data, then the Latitude and Longitude are in Degrees where North and East are Positive. We can get the speed in knots, Direction (Heading) in Degrees where True North equals to zero, and altitude with regards to sea level in meters. We can convert the units as we need. In this way the GPS Data can be collected.

3.3 Getting Information of the Location

Generally the information of a place is stored in an information system as a data referenced by spatial or geographic coordinates. Hence getting the specific available data about a place will not be difficult if we already know the location. We can easily filter the required data based on the coordinates of the location. [11]

3.3.1 Getting locally stored data

As the technology is improving, the devices are getting smaller, smarter, faster and cheaper. So, the limited memory in mobile device will soon be talk of the past. We can assume that storing the data in the device itself will not be a big deal. So, we can use local storage to store the data and display to the user. This will not only increase the processing time to display the data but also reduce the data access cost over the network.

We can create a local database which can store the information about a place. Since our main goal is to provide the destination information in form of image and some description, we can use the folder system to store the images of the particular interest and the relative information on the database. By doing this we can easily access the destination information in the form we want like the description or the picture which is referenced with geographic coordinates. We designed the database to store the latitude, longitude, time, description of the place, image file location.

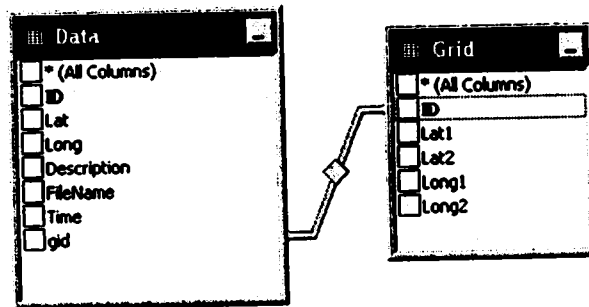


Figure 3.4 (Internal database structure)

The latitude, longitude and time will reference the data to a particular position at a particular time. The time is one factor which should not be missed to reference the data. The information about a place in day time could be something and at night time it could be completely different. For e.g. a footpath in Bangkok at day time could be a nice cloth store, whereas, the exact location in the evening or night time could be a restaurant selling food.

Furthermore, we divide the area into fixed sized grids such that all the data fall under specific grid. By doing this, we can easily find the information we are looking among the huge collection of data. This will help in reduce the search time of the information we are looking for.

3.3.2 Getting data from the network

Another approach to showing the destination information to the user in case there is not enough storage is by using the network and accessing the remote data stored in some server. In trying to do so, we used a flickr API to show the geo-tagged image stored in flickr. Since these images are already referenced by the geographic coordinates, it is easy to filter them and show it to the users. In addition to this, it also has images rated according to the popularity/interesting, relevance, date posted.

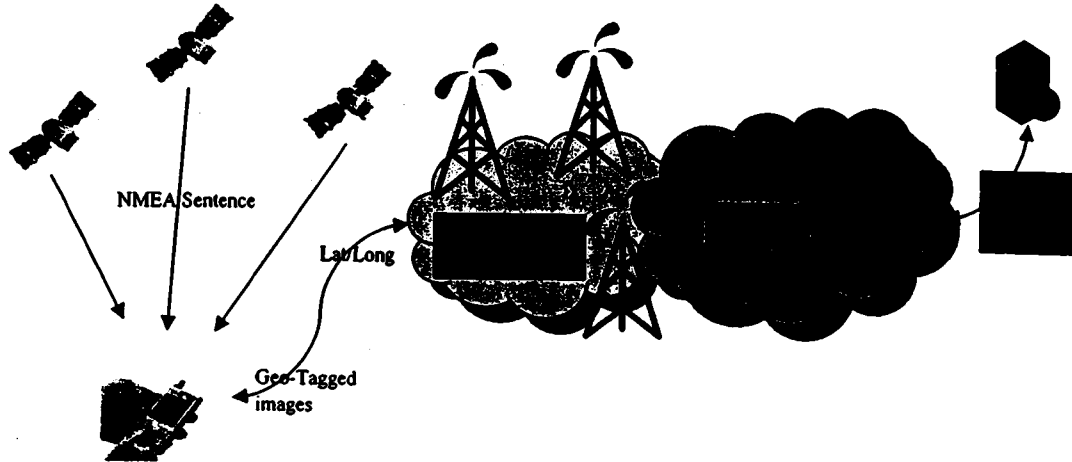


Figure 3.5 (Getting Geo-Tagged Images)

Talking more about flickr, it is an online photo sharing and managing application which provides service like uploading, editing, sharing and geo-tagging the pictures which could be viewed by different users as the author of the picture desires. So when using these services provided by flickr, we extracted the geo-tagged images near our location and showed it to the user. When showing the information, we can choose how much data to be transferred at a request such that it will not slow the transfer of data and at the same time don't have to request for the new data again and again.

3.3.3 Displaying the map and plotting latitude and longitude

Generally, the map that contains the specific information like a tourist spot, road, or pond can be represented using point, line or polygon respectively. This is referred to as vector data. And since we are showing the road map along with some route information and tourist spots that would be suitable to travel; we use a standard vector map showing this. A shape file is a standard vector format image that can store the specific information. We can also add different layers on top of each other to make the data more expressive like a route on top of road network [11, 27].

lot
ne
in
sy
ng

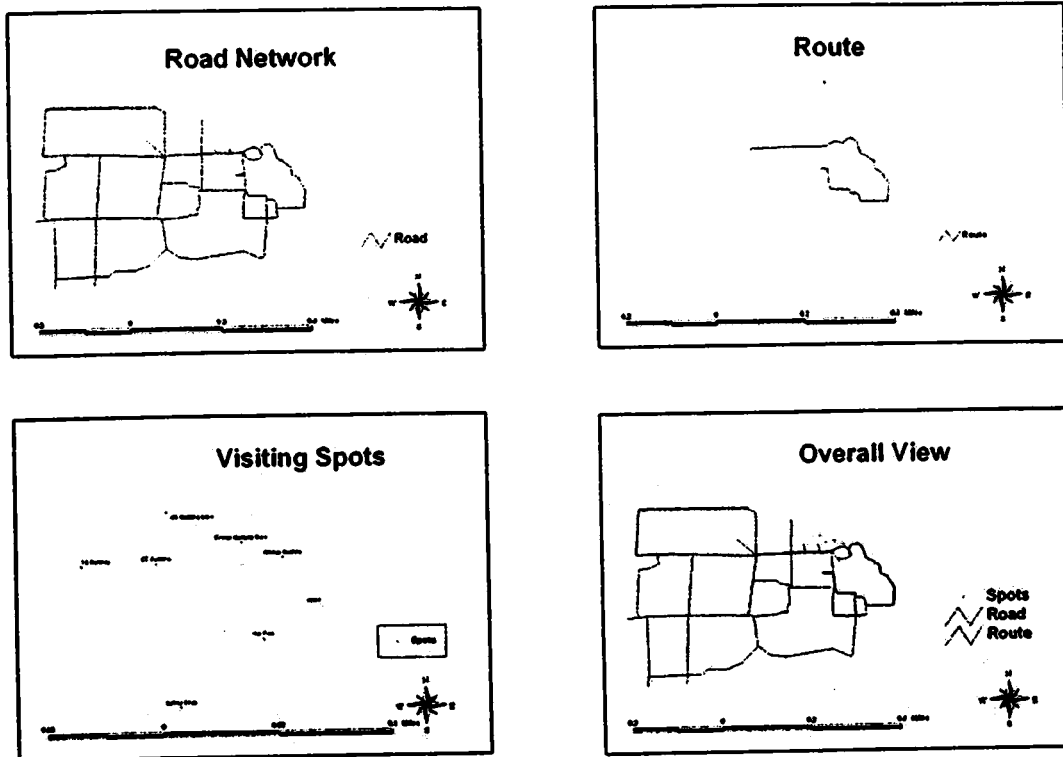


Figure 3.6 (Different data layers)

ich
ld
se
nd
be
me

We can do this using various approaches, and a lot has already been done in this field, so we just use the already available methods and modify them. One of the open source project doing this is SharpMap. It is a mapping library that provides easy access to GIS data and enables users to do spatial querying. It enables us to show the vector data like .shp file and draw extra layers on top of them. It is released under GNU LGPL [28]. So with the help of SharpMap, we can display the map of a particular location with specific information and pinpoint the exact location of the user in the map.

nd
tor
rist
ape
lso
on

We can query the exact location of the user and show it to the user such that it is helpful to them. When giving the information about a place, if the surrounding information is also given, then that might be more helpful when he is travelling, or just for knowledge. A road name, building name, etc can help the user in various ways. So we can add a layer to show the names of the places or relative information on top of the already shown road map. Since we are targeting to plan a travel route to show different places efficiently, we can design different routes map that the user may choose and travel upon it. We can overlay these information one above the other such that the user can get the most out of it.

One thing we need to be aware of is how the data is stored in which coordinate system, datum and projection. Otherwise there is a possibility that they won't overlay each other when we put them together.

Since the earth is not flat and it has to be printed in a flat surface, so people have come up with different approaches to do so. Some of them will preserve distance, some preserve area and so on. SharpMap by default do the equirectangular or "Pseudo-Plate Carré" Projection which project the latitude and longitude data directly to the image by applying a scale. So we just use this projection for the moment, but we can transfer to other projection

if we want. Also, about the coordinate system, we use the latitude/longitude coordinate system or the Geographic coordinate system to represent the data in decimal degrees. And finally the datum that is being used is the WGS84 which is the most popular datum and is used by GPS. [28, 29]

3.4 Finding Shortest Distance and Direction between Two Points

Finding the shortest distance between two points is to draw a straight line between them. However, considering the fact that earth is not a plane, or more precisely, it is exactly neither a sphere nor an ellipsoid, or the fact that the lines of latitude forming a circle are not equal (largest in the equator and decreases as it heads towards poles) makes the calculation of the distance between two points represented in terms of latitude and longitude more difficult. There are different approaches towards calculating the distance between them. The one that we are going to use and which is also a simpler one is the Great Circle Distance which is based on spherical trigonometry. Some other famous methods are: Spheroid Model and Vincenty's Method. [30, 31]

3.4.1 The great circle distance and direction

If (lat1, long1), (lat2, long2) are the two points represented in the Geographic coordinate system, and r is the radius of the sphere, then the distance between them can be calculated using the great circle distance D as:

$$D = r * \arccos(\sin(\text{lat1}) * \sin(\text{lat2}) + \cos(\text{lat1}) * \cos(\text{lat2}) * \cos(\text{long1} - \text{long2}))$$

Equation 3.1

The above formula can give a large round-off error, and since we are expecting the distance that needs to be calculate to be small, we will use haversine's formula which gives better result for smaller distance.

$$D = r * 2 * \arcsin\left(\sqrt{\left(\sin\left(\frac{\text{lat1}-\text{lat2}}{2}\right)\right)^2 + \cos(\text{lat1}) * \cos(\text{lat2}) * \left(\sin\left(\frac{\text{long1}-\text{long2}}{2}\right)\right)^2}\right)$$

Equation 3.2

Here we approximate r to be around 6,371.01 km which is the average radius of the sphere of the earth to calculate the distance. [30-32]

And for the direction from point one to point two; we calculate direction (C) as

$$C = \text{mod}(\arctan2(\sin(\text{long1} - \text{long2}) * \cos(\text{lat2}), \cos(\text{lat1}) * \sin(\text{lat2}) - \sin(\text{lat1}) * \cos(\text{lat2}) * \cos(\text{long1} - \text{long2})), 2 * \pi)$$

Equation 3.3

We will get the direction in terms of angle where we assume the angle below or equal to 15 Degrees and the angle above 345 Degrees to be directed towards north. If it is between 15 and 75 Degrees, then we assume it to be heading towards north-east. If it is between 55 and 105 Degrees, then we assume it to be heading towards east. If it is between 105 and 165 Degrees, then we assume it to be heading towards south-east. If it is between 165 and 195 Degrees, then we assume it to be heading towards south. If it is between 195 and 265

ate
nd
is

Degrees, then we assume it to be heading towards south-west. If it is between 265 and 285 Degrees, then we assume it to be heading towards west. And finally, if it is between 285 and 345 Degrees, then we assume it to be heading towards north-west.

In this way we can calculate the distance and direction between different objects in a real time and use it in various ways as we need it.

m.
tly
are
he
nd
ice
he
ous

ate
ed

)

the
res

)

ere

to
en
55
nd
nd
'65

Chapter 4 Results and Discussion

4.1 Getting Location Position

In this process of creating the application that will not only provide the specific information that is required to travel around a place but also the destination information; we followed a simple example and created an application that would activate the built-in GPS chip in the mobile device. This is the very first and important step accomplished towards fulfilling the goal of providing the user with their location information and its surroundings. After the GPS receiver is activated, it reads the GPS signal containing the exact location information of the user from the satellites. We then parse those signals and use them as per our requirement.

The figure below shows the process of activating the GPS by running the application. For this, we just have to run the GpsSample file inside "Program Files\gps-sample" folder by clicking the file icon as shown in *Figure 4.1*. After this, the application starts and automatically triggers the GPS chip which returns the available data. We then parse the data and extract the data that is useful to us and displays it to user as shown in *Figure 4.2*. Here, the latitude, longitude and the time data are shown. Initially the unit of latitude and longitude is in Decimal Degrees but we convert it to degree minute second and display it, so that we could use it as we want.

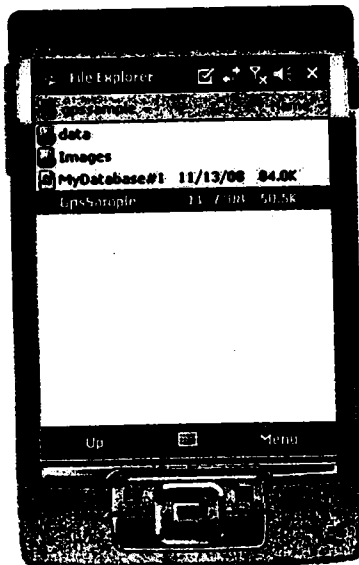


Figure 4.1 (Starting the application)

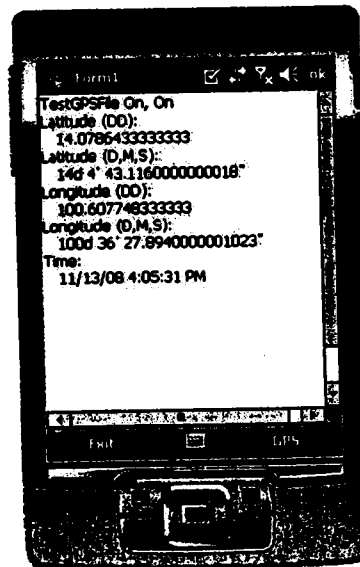


Figure 4.2 (Getting location)

So as soon as we start the application, we get the location data by which we can pinpoint our location. This will not only pinpoint the location but also be used throughout the application where the location information is required. We will be using it in various functionalities like getting the map of the location, getting the destination distance and so on.

4.1.1 The application menu and an approach to save battery life

The first thing that we have to notice in this application is that there are pull-up menus which will provide the users with different functionalities that will fulfill our objectives to make the tour of the individual memorable one. When we click on the GPS menu on the bottom right corner of the screen, then 'Start GPS', 'Stop GPS', 'Destination', 'Nearby Picture', 'Flickr', and 'Map' menu will be displayed. As we start the application, the GPS is already started, so the 'Start GPS' menu is disabled but the rest of the menu is enabled as we start to get the GPS signal.

Since the GPS consumes a lot of battery of the mobile device, so we could stop the GPS by clicking the 'Stop GPS' menu whenever it is not being used. When we do this it will cache the last location information of the device which could be used roughly to estimate the location of the user. When doing this, the 'Start GPS' menu will be enabled while rest of the menus which are dependent on the current location plus the 'Stop GPS' menu itself will also be disabled as shown in *Figure 4.3*.

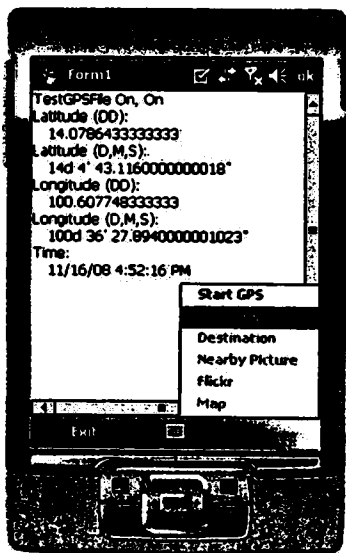


Figure 4.3 (Stopping the GPS)

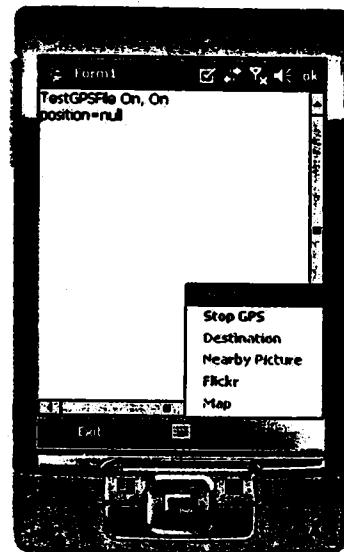


Figure 4.4 (Re-start GPS)

When re-activating the GPS by clicking the 'Start GPS' menu, it will switch the device on, but to get the location information, it needs some time, so in that time, the rest of the menus that are dependent on the location information are disabled. But as soon as the location information is available in terms of latitude and longitude, then the rest of the menus will also be enabled. This is illustrated in *Figure 4.4*.

4.2 Getting Information of the Location

In this application, we tried to get the information of that location in two different approaches. In the first approach, we get the information that is stored in the local database and is referenced by the geographic coordinate. In this approach, we did some calculations that could be useful to the user like the destination distance and direction. In the second approach, we get the information of the place via web. In this approach we use the web service and extract the geo-tagged information in terms of images and show it to the user.

4.2.1 Getting locally stored data

Our main objective of facilitating the user by showing the destination information was accomplished using the 'Destination' menu. When we click this menu, then it will show us the images of the destination along with the optional information like description, distance and direction from the current location. As shown in *Figure 4.5*, when the location data is available, then the 'Destination' menu is enabled. When we select it, then it will show us the destination thumbnail images sorted according to the distance along with the signal status from the satellite as shown in *Figure 4.6*.

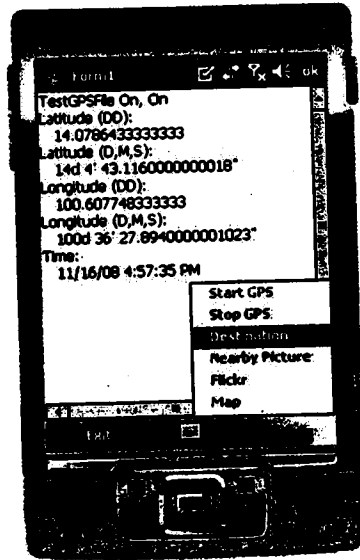


Figure 4.5(Destination Menu selection)

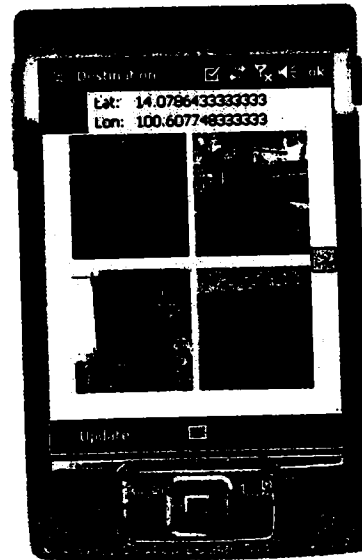


Figure 4.6(Destination image information)

The top-left image in *Figure 4.6* represents the type of fix like 2-D or 3-D between the receiver and the satellite. Also the user can know whether the satellite signal is available or not by looking at the lat/lon data available on top of the screen. Then the destination images are shown as thumbnails in the screen where the nearest destination from the current location is shown at the top then right then bottom. If there are more destination images, then a button will be visible on the middle-right of the screen and middle-left of the screen through which we can navigate forward and backward.

When we want to see the detail picture of a destination, then we just have to select the thumbnail and then it will show us the full picture of the destination as shown in *Figure 4.7*. And then if we want to see the details of the destination then we just have to click on

the destination picture and then it will show us the description, distance and direction from the current location separated with semicolon as shown in *Figure 4.8*. If we again click on the image then it will take us back to the thumbnail screen where we can choose other destinations and view its detail.

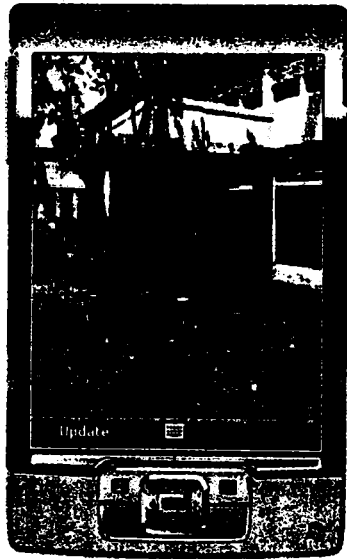


Figure 4.7 (Enlarged destination picture)

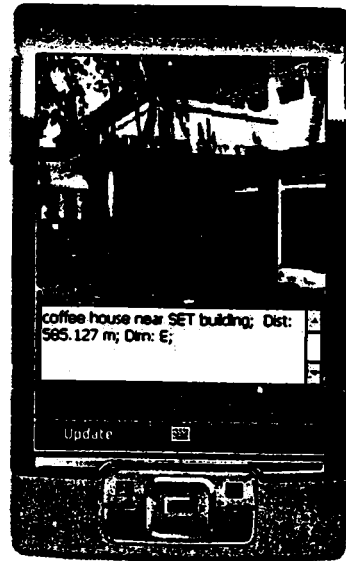


Figure 4.8 (Destination with description)

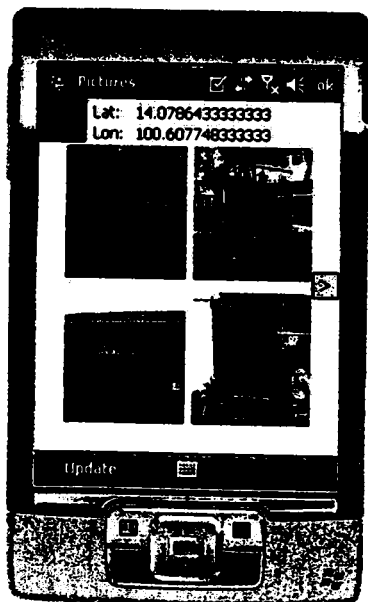


Figure 4.9 (Nearby Images)

Another important feature that could be useful to the user is showing the nearby images from the user's location such that the user could visually identify where he is and at the same time see different interesting locations which might be in his interest and is nearby to him. This feature will also show him the destination information along with distance and direction as in Destination function. *Figure 4.9* illustrates this function where the user gets to preview the nearby images and at the same time see its information as in *Figure 4.7* and *Figure 4.8*.

4.2.2 Getting data from the network

The alternative way of showing the user with the surrounding information via the web was accomplished using the Flickr API. When the location information in terms of latitude and longitude is available, then the 'Flickr' menu in the main form is activated. When we click it, then it will get us the geo-tagged images from Flickr's server. Only after we click the 'Update' menu in the bottom-left corner of the screen, then the flicker

n
n
r

pictures will be updated as the location changes. This can be seen in *Figure 4.10* and *Figure 4.11*.

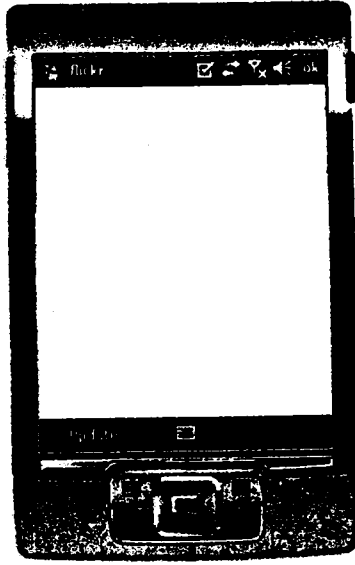


Figure 4.10 (Accessing data from internet)

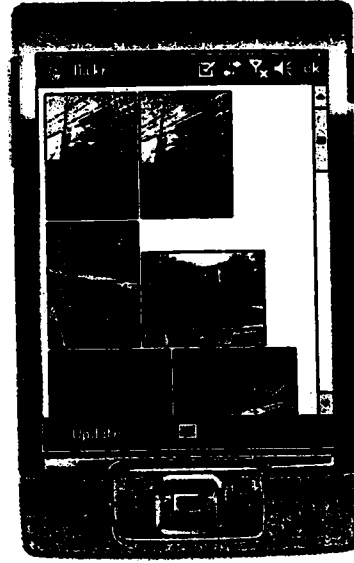


Figure 4.11 (Nearby geo-tagged Flickr images)

to
re
ly
re
is
so
th
n.
er
re
re

re
as
re
id
re
it
's
in
er

4.3 Showing Globally Referenced Local Position

After we get the location information in terms of latitude and longitude, then the Map menu in the main form is activated. When we click the 'Map' menu then we get the globally referenced local position of the user. In *Figure 4.12*, the Red circle with x in the center of the screen shows the globally referenced local position of the user inside the AIT. The other square dots are the surrounding information from where the user stands and he/she can view in details of it in terms of description, distance and direction. The lines are the road that can be used to travel to different places as desired. And finally, the small point on top is the direction indicator which will move as we move. If we are heading north then it goes to center-top of screen, if east, then it goes to center-right position and so on.

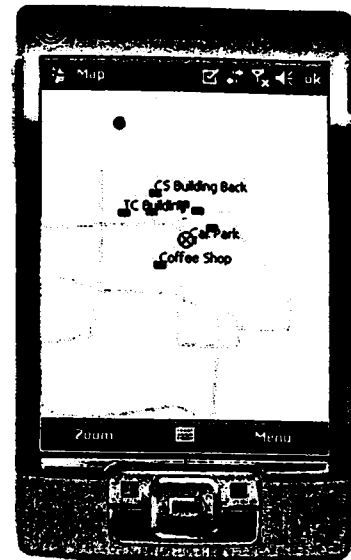


Figure 4.12 (Globally referenced local position)

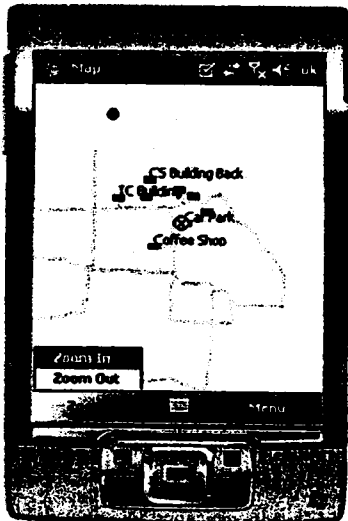


Figure 4.13 (Zooming feature)

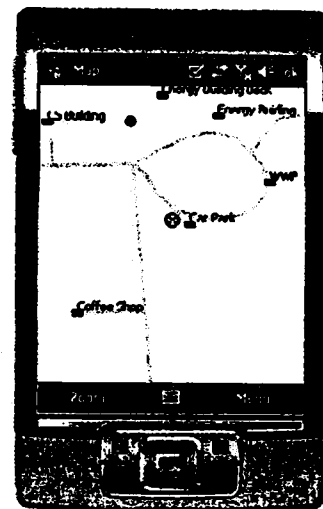


Figure 4.14 (Zoomed in map)

We can also zoom into the specific location as we desire by clicking the zoom menu on the bottom-left of the screen. When we click it then we can either zoom-in or zoom-out as we desire. *Figure 4.13* and *Figure 4.14* shows the zooming feature where the user is zooming into a specific location.

Another objective of this application was to show the user the desirable path to travel to the destination. We could use various approach to do so. Here we have used an approach where the user will be able to load the desirable route to travel in order to reach the destination. This route at the moment is static but later can easily be converted to dynamic approach where the destination route will be automatically calculated. When we click the

Menu in the bottom-right corner of the screen of the Map interface then we can select 'Load Route' menu which will load the route to travel to the destination. When we load the route, then it will overlap with the existing road map of that place and the only difference will be the color of the road that the user is supposed to follow. As seen in *Figure 4.15* and *Figure 4.16*, the user selects the load route and then the route is loaded in the existing map with green color. The user has to follow the route which he can easily do because his globally referenced local position is also shown in the map.

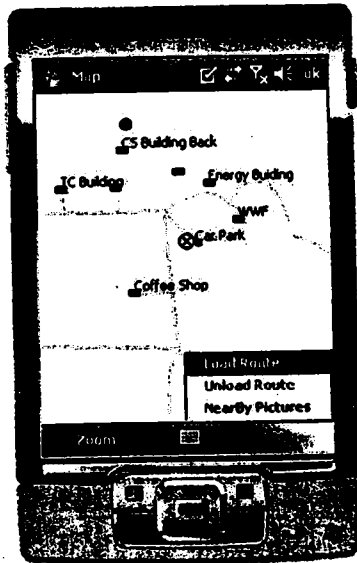


Figure 4.15 (Loading the travel route)

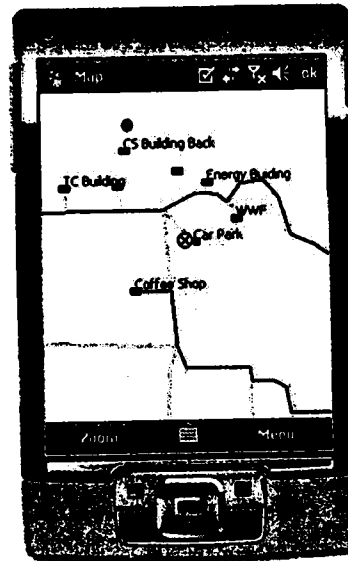


Figure 4.16 (Route loaded as green color line)

Looking at all of these results we can say that all of the objectives have been fulfilled in one way or the other. There could have been different approaches to obtain these results but we choose the simplest ones and got the desired results which helped in proving the concept that the destination information in terms of image could be a valuable asset in the mobile tourist system.

ne
ve
ig

to
ch
he
ic
he

Chapter 5

Conclusion and Recommendations

5.1 Conclusion

In conclusion, we tried to build an application that would show the importance of destination information and various ways to access it via handheld GPS enabled mobile devices. When we provide the destination information in terms of description and images along with the time specific information, it helped to remove any confusion of the traveler's tour. He knew his exact location, where to go next, how to go there, and what to expect there at different time.

In this process, we built an application that activated the built-in GPS receiver and extracted the location information in terms of latitude and longitude. We then used this information to extract the geographically referenced destination information. Since Standalone approach towards data access could be considered as a base case, so we used it to get the destination information. Later we even tried to access the destination information via the web based approach which when combined with standalone approach and then with slight modification could result in the hybrid approach.

When providing the destination information, we had to do some calculation regarding the distance between two objects and its direction. We used the famous great-circle distance and bearing to do this, and showed the time specific, location dependent information to the user that could be helpful for the travel.

So at the end we can say that the destination information in terms of personalized visual image along with time based knowledge and information could be a valuable asset in the mobile tourist system.

5.2 Recommendations and Future Work

In this fast developing world of mobile internet and device technology, we can expect a lot to change in a very little time. There is this 3G network system which was not long ago introduced for high speed data transfer and internet connection via mobile, the researchers are already studying and developing the 4G system which is expected to have better performance, connection speed and reliability. And then there is WiMax; a broadband wireless access that permits users to access the high speed internet and data transfer. With these technologies ahead of us plus other emerging technologies like Multihop relay networks coming soon into this field, we could expect something like centralized data storage system where all the information will be stored in a central server, and the user will access or update it via the internet.[33]

We could fully utilize these upcoming technologies plus some other web services that could raise the standard of this application to next level. We could, for example, use the application like Microsoft's Photosynth and show the user the overall view of the location in 3D model and at the same time allowing user to see the specific information. Or we could use some voice guided tour in the background and show the destination information in form of pictures. We could even allow the user to update their own personalize information which could be shared among their friends or contribute to the overall community.

Furthermore, there are other satellite navigation systems under development. Systems like Beidou from China, Galileo from European Union and European Space Agency, GLONASS from Russia with partnership with India, and other proposed systems when starts to operate, they will surely bring the current accuracy of the satellite navigation to few centimeters [16]. Not only this, there is AGPS which is promising a fast first fix to satellite and higher accuracy even inside a room. With this accuracy we could further more personalize the system and may be even put the pictures of the shop owner and information about them so that user could be more comfortable around that place.

There are similar and many more ideas that could be implemented in this application to shape the future of mobile tourism and walking tour system.

References

1. Tan, E. M., Goh, D. H., Theng, Y., and Foo, S.,(2007). An analysis of services for the mobile tourist. In *Proceedings of the 4th international Conference on Mobile Technology, Applications, and Systems and the 1st international Symposium on Computer Human interaction in Mobile Technology* (Singapore, September 10 - 12, 2007). Mobility '07. ACM, New York, NY, 490-494. DOI=. Retrieved 2008 from <http://doi.acm.org/10.1145/1378063.1378142>
2. Wang, J.J., Song, Z., Lei, P. and Sheriff, R.E.,(2005). "Design and Evaluation of M-Commerce Applications," *Communications, 2005 Asia-Pacific Conference on* , vol., no., pp. 745-749, 03-05 Oct. 2005. Retrieved 2008 from URL: <http://ieeexplore.ieee.org/iel5/10412/33074/01554161.pdf?isnumber=33074&arnumber=1554161&arnumber=1554161&arSt=+745&ared=+749&arAuthor=Wang%2C+J.J.%3B++Song%2C+Z.%3B++Lei%2C+P.%3B++Sheriff%2C+R.E>
3. TURBAN KING, MCKAY, LEE, VIEHLAND,(2008). *Electronic Commerce 2008 A Managerial Perspectives*; 2008
4. Squidoo,(2008). What is M-Commerce. Retrieved 2008 from URL: <http://www.squidoo.com/aboutmcommerce>
5. Das, R., Wang, J.J. and Lei, P.,(2005). " A socio-cultural analysis of the present and the future of the m-commerce industry", *Handbook of research in mobile business:technical, methodological and social perspective*, editor B. Unhelkar, IDEA Group Inc., 2005
6. Jagoe, A.,(2002). *Mobile Location Services, The Definitive Guide*. Prentice Hall; 2002
7. Norrie, M. C., Signer, B., Grossniklaus, M., Belotti, R., Decurtins, C., and Weibel, N.,(2007). Context-aware platform for mobile data management. *Wirel. Netw.* 13, 6 (Dec. 2007), 855-870. DOI=. Retrieved 2008 from <http://dx.doi.org/10.1007/s11276-006-9858-y>
8. Chalmers, M. and Brown, B.,(September 2003). *Tourism and mobile technology. Proceeding of Conference on computer Supported Cooperative Work*, Kluwer Academic Press.
9. Hinze, A. and Buchanan, G.,(September 2005) *Context-awareness in Mobile Tourist Information Systems: Challenges for User Interaction*. Proceeding of conference on mobile HCI.
10. CMT,(2008). *Introduction to the Global Positioning System for GIS and TRAVERSE*. Retrieved 2008 from URL: <http://www.cmtinc.com/gpsbook/index.htm>
11. Dr. Tripathi, Nitin K.,(January 2007). *Lecture notes, Course No. AT76.01. Bangkok: Asian Institute of Technology*

12. How Stuff Works,(2008). How GPS receivers work. Retrieved 2008 from URL:
<http://www.howstuffworks.com/gps.htm>
13. Mobile Tech Review,(2008). GPS FAQ. Retrieved 2008 from URL:
http://www.mobiletechreview.com/tips/GPS/GPS_faq.htm
14. Trimble,(2008). GPS Tutorial. Retrieved 2008 from URL:
<http://www.trimble.com/gps/>
15. WillKommen,(2008). GPS – explained. Retrieved 2008 from URL:
<http://www.kowoma.de/en/gps/>
16. Wikipedia,(2008). Global Positioning System. Retrieved 2008 from URL:
http://www.en.wikipedia.org/wiki/Global_Positioning_System
17. GPS Lodge,(2008). What is SiRF Star III chipset?. Retrieved 2008 from URL:
<http://www.gpslodge.com/archives/004566.php>
18. Wikipedia,(2008). NMEA 0183. Retrieved 2008 from URL:
http://en.wikipedia.org/wiki/NMEA_0183
19. Mehaffey, J., Yeazel, J., Penrod, S. and Deiss, A.,(2008). NMEA data. Retrieved 2008 from URL: <http://www.gpsinformation.org/dale/nmea.htm>
20. Betke, K.,(2001). NMEA Description; May 2000. Revised August 2001
21. Wikipedia,(2008). Mobile development. Retrieved 2008 from URL:
http://en.wikipedia.org/wiki/Mobile_development
22. Andrew R. Hickey,(2008). Mobile platforms: J2ME -- looking for its place in the enterprise. Retrieved 2008 from URL:
http://searchmobilecomputing.techtarget.com/news/article/0,289142,sid40_gci1225396,00.html
23. redFivelabs,(2008) . .NET Compact Framework for Symbian OS. Retrieved 2008 from URL: <http://www.redfivelabs.com/>
24. Person, J.,(2008). Writing Your Own GPS Applications. Retrieved 2008 from URL: <http://www.codeproject.com/KB/mobile/WritingGPSApplications1.aspx>
25. Maarten,(2008). Windows Mobile Location Awareness Webcast. Retrieved 2008 from URL:
<http://www.dotnetfordevices.com/forum.html?monthidx=7&yearidx=2008>
26. Windows Mobile Developer Center,(2008). Windows Mobile. Retrieved 2008 from URL: <http://msdn.microsoft.com/en-us/windowsmobile/default.aspx>
27. Google Map,(2008). Google Maps for your phone. Retrieved 2008 from URL:
<http://www.google.com/mobile/gmm/index.html>
28. Sharp Map,(2008). Geospatial application framework for the CRL. Retrieved 2008 from URL: <http://www.codeplex.com/SharpMap/Wiki/View.aspx>

29. Morten,(2008). Spherical/Web Mercator: EPSG code 3785. Retrieved 2008 from URL: <http://www.sharpgis.net/category/GIS.aspx>
30. Williams, E.,(2008). Aviation Formulary V1.44. Retrieved 2008 from URL: <http://williams.best.vwh.net/avform.htm>
31. Geoscience Australia,(2008). Great Circle Distance. Retrieved 2008 from URL: <http://www.ga.gov.au/geodesy/datums/distance.jsp#circle>
32. Wikipedia,(2008). Great-circle distance. Retrieved 2008 from URL: http://en.wikipedia.org/wiki/Great_circle_distance
33. Haskin, D.,(2008). 13 future mobile technologies that will change your life. Retrieved 2008 from URL: <http://www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=9043245&pageNumber=1>

5

n

3

Appendix B

Code to Display the Location Information (Sample Modified from Windows Mobile 6 SDK)

```
if (gps.Opened)
{
    string str = "";
    if (position != null)
    {
        if (position.LatitudeValid)
        {
            str += "Latitude (DD):\n " + position.Latitude + "\n";
            str += "Latitude (D,M,S):\n " + position.LatitudeInDegreesMinutesSeconds +
"\n";
        }

        if (position.LongitudeValid)
        {
            str += "Longitude (DD):\n " + position.Longitude + "\n";
            str += "Longitude (D,M,S):\n " + position.LongitudeInDegreesMinutesSeconds
+ "\n";
        }

        if (position.SeaLevelAltitudeValid)
        {
            str += "Sea-Level Altitude:\n" + position.SeaLevelAltitude + " m\n";
        }

        if (position.SatellitesInSolutionValid && position.SatellitesInViewValid &&
position.SatelliteCountValid)
        {
            str += "Satellite Count:\n " + position.GetSatellitesInSolution().Length + "/" +
position.GetSatellitesInView().Length + " (" +
position.SatelliteCount + ")n";
        }

        str += "Time:\n " + DateTime.Now.ToString() + "\n";

        if (position.LatitudeValid && position.LongitudeValid)
        {
            this.ShowImageMenuItem.Enabled = true;
            this.FlickrMenuItem.Enabled = true;
            this.menuItem1.Enabled = true;
            this.DestinationMenuItem.Enabled = true;
            if (position.HeadingValid)
                shp.movePoint(position.Latitude, position.Longitude, position.Heading);
            else
                shp.movePoint(position.Latitude, position.Longitude, 400);
        }
    }
}
```

```
    }  
    else  
    {  
        this.ShowImageMenuItem.Enabled = false;  
        this.FlickrMenuItem.Enabled = false;  
        this.menuItem1.Enabled = false;  
        this.DestinationMenuItem.Enabled = false;  
    }  
}  
  
status.Text = str;  
  
if (position == null)  
    status.Text += "position=null\n";  
}
```


Code to Access Database and Calculate Distance and Heading

```
private SqlCeConnection _conn;
double lat1, long1;
int time, dest=1;

public SqlCeResultSet Pict()
{
    if (Form1.position != null)
    {
        lat1 = Form1.position.Latitude;
        long1 = Form1.position.Longitude;
        if (_conn == null)
            _conn = new SqlCeConnection(@"Data Source ="
                +
                (System.IO.Path.GetDirectoryPath(System.Reflection.Assembly.GetExecutingAssembly()
                    ).GetName().CodeBase) + "\\MyDatabase#1.sdf;")
                + "Password = 'iwts'");

        SqlCeCommand cmd = new SqlCeCommand();
        cmd.Connection = _conn;

        if (System.DateTime.Now.Hour > 18 || System.DateTime.Now.Hour < 4)
            time = 1;
        else
            time = 0;

        cmd.CommandText = "SELECT GridID, Lat, Long, Image, img, Distance,
            Direction, Angle, Time, ID FROM Image WHERE Image.GridID IN ( SELECT
            Grid.GridID FROM Grid WHERE Grid.Lat1 <= " + lat1 + " AND Grid.Lat2 >= " + lat1 +
            " AND Grid.Long1 <= " + long1 + " AND Grid.Long2 >= " + long1 + " ) AND
            Image.Time = "+time+" ORDER BY Distance ASC";

        if (_conn.State == ConnectionState.Closed)
            _conn.Open();

        SqlCeResultSet resultSet1 = cmd.ExecuteResultSet(ResultSetOptions.Scrollable |
            ResultSetOptions.Updatable);

        if (resultSet1.HasRows.Equals(true))
        {
            Double x, y, z, a, b, c = 0, dlat, dlong; //c=direction, z=distance.

            while (resultSet1.Read())
            {
                dlong = ((double)resultSet1.GetDouble(2) - long1) * Math.PI / 180;
                dlat = ((double)resultSet1.GetDouble(1) - lat1) * Math.PI / 180;
            }
        }
    }
}
```

```

x = Math.Pow(Math.Sin((dlat) / 2), 2);
y = Math.Cos(lat1 * Math.PI / 180) *
Math.Cos((double)resultSet1.GetDouble(1) * Math.PI / 180) * Math.Pow(Math.Sin(dlong
/ 2), 2);
z = 2 * Math.Asin(Math.Sqrt(x + y));

b = Math.Sin(dlong) * Math.Cos((double)resultSet1.GetDouble(1) *
Math.PI / 180);
a = Math.Cos(lat1 * Math.PI / 180) *
Math.Sin((double)resultSet1.GetDouble(1) * Math.PI / 180) - Math.Sin(lat1 * Math.PI /
180) * Math.Cos((double)resultSet1.GetDouble(1) * Math.PI / 180) * Math.Cos(dlong);
if (b > 0)
{
    if (a > 0) c = 180 / Math.PI * Math.Atan(b / a);
    if (a < 0) c = (180 - 180 / Math.PI * Math.Atan(-b / a));
    if (a == 0) c = 90;
}
else if (b < 0)
{
    if (a > 0) c = 360 - 180 / Math.PI * Math.Atan((-b) / a);
    if (a < 0) c = 180 / Math.PI * Math.Atan(b / a) + 180;
    if (a == 0) c = 270;
}
else
{
    if (a > 0) c = 0;
    if (a < 0) c = 180;
    if (a == 0) c = 0;//2 points are same
}

if (c > 345 || c <= 15)
    resultSet1.SetString(6, "N");
else if (c > 15 && c <= 75)
    resultSet1.SetString(6, "NE");
else if (c > 75 && c <= 105)
    resultSet1.SetString(6, "E");
else if (c > 105 && c <= 165)
    resultSet1.SetString(6, "SE");
else if (c > 165 && c <= 195)
    resultSet1.SetString(6, "S");
else if (c > 195 && c <= 265)
    resultSet1.SetString(6, "SW");
else if (c > 265 && c <= 285)
    resultSet1.SetString(6, "W");
else if (c > 285 && c <= 345)
    resultSet1.SetString(6, "NW");

```

```
        resultSet1.SetDouble(5, Math.Round(6371010 * z, 3)); //1852 * 180 /  
Math.PI * 60 * z //distance in meters  
        resultSet1.SetDouble(7, c);  
  
        resultSet1.Update();  
    }  
    }  
    resultSet1.Dispose();  
    resultSet1 = cmd.ExecuteNonQuery(ResultSetOptions.Scrollable |  
ResultSetOptions.Updatable);  
    return resultSet1;  
}  
else  
    return null;
```

Adding Different Layers of Information using SharpMap

```
myMap = new SharpMap.Map(new Size(this.Width, this.Height));
myMap.MinimumZoom = 0.0001;
myLayer = new SharpMap.Layers.VectorLayer("Road");
routLayer = new SharpMap.Layers.VectorLayer("Rout");
PointLayer = new SharpMap.Layers.VectorLayer("PointLayer");
labelLayer = new SharpMap.Layers.VectorLayer("labelLayer");
labelPoint = new SharpMap.Layers.LabelLayer("labelPoint");
```

```
Collection = new SharpMap.Geometries.GeometryCollection();
GeomColl = new System.Collections.Generic.List<SharpMap.Geometries.Geometry>();
```

```
myLayer.DataSource = new SharpMap.Data.Providers.ShapeFile("Program
Files\\gpssample\\data\\newroad.shp");
myLayer.Style.Line.Width = 3;
myLayer.Style.Line.Color = Color.Yellow;
myLayer.Style.Fill = new SolidBrush(Color.Transparent);
myLayer.Style.EnableOutline = true;
myLayer.MaxVisible = 4000;
```

```
myMap.Layers.Add(myLayer);
myMap.Center = new SharpMap.Geometries.Point(100.613343333333,
14.0801633333333);
```

```
labelLayer.DataSource = new SharpMap.Data.Providers.ShapeFile("Program
Files\\gpssample\\data\\label.shp");
labelLayer.Style.Symbol = new Bitmap("Program Files\\gpssample\\Images\\icon.png");
labelLayer.Style.Fill = new SolidBrush(Color.Transparent);
```

```
labelLayer.MaxVisible = 50;
myMap.Layers.Add(labelLayer);
```

```
labelPoint.DataSource = labelLayer.DataSource;
labelPoint.Enabled = true;
labelPoint.LabelColumn = "Label";
labelPoint.Style = new SharpMap.Styles.LabelStyle();
labelPoint.Style.ForeColor = Color.Black;
labelPoint.Style.Font = new Font(FontFamily.GenericSerif, 8, FontStyle.Regular);
labelPoint.MaxVisible = 50;
labelPoint.Style.HorizontalAlignment =
SharpMap.Styles.LabelStyle.HorizontalAlignmentEnum.Center;
labelPoint.Style.VerticalAlignment =
SharpMap.Styles.LabelStyle.VerticalAlignmentEnum.Top;
labelPoint.Style.Halo = new Pen(Color.Yellow, 2);
labelPoint.Style.CollisionDetection = true;
```

```
myMap.Layers.Add(labelPoint);
```

```
point = new SharpMap.Geometries.Point(100.613343, 14.08026);
```

```
GeomColl.Add(point);
```

```
PointLayer.DataSource = new SharpMap.Data.Providers.GeometryProvider(GeomColl);
```

```
myMap.Layers.Add(PointLayer);
```

```
myMap.Zoom = 0.0025; //Set zoom level
```

```
this.pictureBox2.Image = myMap.GetMap();
```