Keio University Master's Thesis Academic Year 2007

Burari-navi: QoS-driven Exploratory-walk Navigation System

Keio University Graduate School of Media and Governance

Hidetoshi Tokuda

Abstract

Burari-navi: QoS-driven Exploratory-walk Navigation System

Summary

This thesis purposes “Burari-navi”, a Quality-of-Satisfaction (QoS)-driven exploratory-walk navigation system which enables users to find an interesting spot to visit, walk with guidance, and enjoy without any verbal input of desire.

Currently, navigation systems are installed on mobile devices and becoming widely used. Navigation systems are systems that support users to find, visit, and enjoy spending a time at a specific shop, restaurant, and any specific locations that users are interested in. And recently, a navigation system for pedestrian has been a hot topic in both research communities and the industrial market.

In this thesis, we focus on pedestrians using navigation systems and introduce a new category of users called “exploratory-walk users”. Users will be categorized into two types: purposive-walk users whom has an explicit verbal desire and exploratory-walk users whom do not have a verbal desire but has highly abstracted potential desire to do “something”. Unfortunately, exploratory-walk users cannot use existing services because existing services require users’ verbal purpose at the entrance of the system and/or also the choices are not purposed to the users in an appropriate style.

We purpose a navigation system designed for exploratory-walk users called “Burari-navi”. Burari-navi only requires a length of time-slot and a need of return at the entrance of the system to enable exploratory-walk users to browse choices without a verbal purpose input. Burari-navi provides a large number of choices to the user efficiently using highly brows-able Surface Browsing User Interface, relevance based choice sorting, and history based re-sorting. Burari-navi gives a non-dropped choices to the user to avoid accidently cutting off the user’s needs via a user interface because the user’s needs are ambiguous and complicated to predict. And the large number of choices will be purposed to the user via a Surface Browsing User Interface with high brows-ability which covers the risk of oversight caused by dilution and avoids overwhelming the user. Then, Burari-navi sorts choices in order of relevance based on time, spot attribute, and transition probability to decrease the mean cost of discovery and searching from choices. In addition, Burari-navi re-sorts choices based on the user’s history to avoid redundant recommendation such as repeating recommendation of restaurants.

We implemented a prototype of Burari-navi and demonstrated the needs and effectiveness of the Burari-navi system by evaluating the system from two perspectives: users’ Quality-of-Satisfaction and system usability.

Keywords:

1 Navigation System   2 Browsing User Interface   3 Recommendation System

4 Location-aware System   5 Ubiquitous Computing

Keio University Graduate School of Media and Governance

Hidetoshi Tokuda

修士論文要旨　　　2007 年度 (平成 19 年度)

ぶらりナビ：
QoS 主導型散策ナビゲーションシステムの構築

論文要旨

位置情報取得機能を組み込んだ携帯端末の急速な普及により，利用者の位置や状態に合わせた情報・サービスの提供を行うロケーションアウェアシステムが広く注目されている．また，利用者の生活中におけるナビゲーションシステムのサポート可能なシーンが拡大している．これまでナビゲーションシステムは，誘導を主目的としたシステムの事を指していた．しかし，ナビゲーションシステムとは利用者の欲求認識から行動完了までの一部，または全てをサポートするシステムをナビゲーションシステムとする．

近年，最も注目されているのが歩行者ナビゲーションシステムである．既存の歩行者ナビゲーションシステムは，携帯端末上での移動のサポート，目的に沿った目的地の決定支援，地図上での情報共有を可能にしている．歩行者ナビゲーションシステムの利用者には目的志向型の利用者と散策志向型の利用者があり、目的志向型は利用者自身が目的および欲求を明確に提示可能な状態を，散策指向型は言語化可能な目的，欲求は持たずあいまいな潜在的欲求を保持した状態を指す．既存のナビゲーションシステムは目的志向型利用者を対象としており，散策指向型の利用者をサポートしない．

本論文では，散策指向型利用者を対象とした “Quality-of-Satisfaction (QoS)” 主導型散策ナビゲーションシステム「ぶらりナビ」を提案する．ぶらりナビは利用者に対して明確な目的や欲求の入力を要求せず，散策時間および帰着の必要性の入力のみで利用者に対して到達可能な候補地を提示する事で潜在的欲求を引き出し，目的地の決定および候補の特定をサポートするシステムである．ぶらりナビは利用者に対して提示する候補数を絞らず一覧性の高いサーフェスブラウジングインタフェース，候補の整列，利用履歴を基にした再整列を行う事で利用者に効率の良い提案を実現している．また，散策指向型利用者の特徴のひとつである目的地の突然の変更や，候補地の突然の再探索に対応している．そして，歩行者向けの誘導システムとの連携により一貫したサポートを実現している．

ぶらりナビのユーザインタフェースを実現したプロトタイプを実装し，シミュレーション実験を行い，散策指向型利用者の行動サポートの実現度，およびユーザビリティの評価を行うことで, ぶらりナビ有効性を検証した．

キーワード

1 ナビゲーションシステム　2 ブラウジングユーザインタフェース
3 レコメンデーションシステム　4 ロケーションアウェアシステム
5 ユビキタスコンピューティング

慶應義塾大学大学院 政策・メディア研究科
徳田 英隼

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter introduces and describes the field of exploratory-walk navigation systems.

1.1 Background and Motivation

Currently, "Location-aware service" is becoming a big trend in a society [1, 2]. Location aware service is a service that provides useful information to a user based on locational information such as current location, activity history, locational relations between other users, and more. Navigation system is a most popular location aware service that is used in our daily life. Navigation systems used to be installed on cars, and recently navigation systems are being installed on mobile devices like cell phones and PDA [3, 4]. Japan has a leading market of mobile navigation systems and there are several successful implementations: EZ Navi Walk [3], NAVI TIME [4], and Everyone's Map [5].

As the platform for the navigation system has been changing, also the coverage of the system in our life has been increasing [6]. In the past few years, the only coverage of the navigation system was route guidance. But first, the system started to extend its coverage from route guidance to area information service and trip planner. And now, the system is not only for cars but also pedestrians with mobile devices so that the user can receive rich information even while walking. We have been focusing on a pedestrian navigation systems since the year of 2006, and have implemented some prototypes [7] to demonstrate the effectiveness and to gain constructive comments on it.

Since there are various types of platforms and huge coverage of service, it is becoming more difficult and insufficient to build an almighty system that could support everything. A N-to-N mapping which is "many services for many types of usage" is becoming a standard in the market. And many kinds of services are arising rapidly [6]. But on the other hand, still there are many scenes that are not supported by any systems or services yet.

"Exploratory-walk" is an important style of activities in our daily life that occurs frequently but still not supported by navigation system. Exploratory-walk is a situation that a user does not have any verbal purpose or requirement but has an amount of time needs to be filled with some interesting activities. An obvious example of an occurrence of the exploratory-walk is when a person became free for an hour because the meeting was cancelled right before the time.

In this case, the user is free for an hour, want to enjoy and do not waste, but has no specific plan to do at the time. One possible solution for this situation will be to visit a bookstore and thumb-through few guidebooks that look interesting. But this will fail if the user does not know where the book store is and also will need to take a walk to the bookstore. Most practicable and convenient solution will be to retrieve some interesting information to choose next place to visit. Existing services could provide such information only when a user has a verbal desire or a request to input and that is difficult in "Exploratory-walk".

As explained above, it is clear that exploratory walk is not supported well by existing services and an optimized system for an exploratory-walk is desired by the society.

1.2 Research Goal

A research goal of this thesis is to construct an "exploratory-walk navigation system" which is a navigation system that supports exploratory-walk user model by providing information to users without a verbal input of desire or a request. By using exploratory-walk navigation system, users should be able to browse, select, and visit spots and sites anytime when they need without any problems. In this thesis, we create and clarify a new category of users and its characteristic: exploratory-walk users. And then design, implement, and demonstrate the effectiveness of an exploratory-walk navigation system called "Burari-navi".

1.3 Thesis Organization

This thesis is organized as follows. Chapter 2 introduces and then clarifies the definition of an Exploratory-walk user model and its characteristics. Chapter 3 clarifies the requirements for the Exploratory-walk navigation system. Chapter 4 purposes an exploratory-walk navigation system called "Burari-navi" and describes its concept. And then, at Chapter 5, we focus on architecture and each component of Burari-navi at the level of implementation. Chapter 6 evaluates Burari-navi from the perspective of meeting requirements and usability. Chapter 7 introduces past and some on-going related works to clarify differences and strengths of Burari-navi. Finally, Chapter 8 summarizes this thesis by focusing on efforts made by Burari-navi and future works of it.

Chapter 2

Exploratory-walk User Model

In this chapter, we define and describe an "exploratory-walk user model" which is a new category of navigation system users. First, we describe the overview of the exploratory-walk by comparing with a purposive-walk. Next, we discuss deeply about the exploratory-walk and its characteristics. And finally, we focus on how exploratory-walk users evaluate a Quality-of-Satisfaction level and what do they need as support from navigation systems.

2.1 Purposive-walk and Exploratory-walk

Users of navigation systems could be modeled into two types of models. One is a user in a purposive-walk and the other is a user in an exploratory-walk. In this section, we define and describe the overview of each user model by comparing each other from two aspects: existence of purpose and stability of context.

At a glance, purposive-walk users have a clear purpose for their activity and therefore have stable context. On the other hand, exploratory-walk users do not have an obvious purpose and their context such as interests and desire could be changed easily from external factors. Table 2.1 shows the difference of purposive-walk user model and exploratory-walk user model. We explain each user model at following Section 2.1.1 and Section 2.1.2.

Table 2.1: User Model of Navigation System

Type	Existence of Purpose	Context Stability
Purposive Walk	Yes	Stable
Exploratory Walk	**No**	**Unstable**

2.1.1 Purposive-walk

Purposive-walk is a common type of users of existing navigation systems. Purposive-walk users own a purpose, desire, or a request for their activity. A simple example could be a user who is looking for the nearest bookstore to get a book. Or, it could be a user who is hungry and wants to get something to eat. The first user's purpose is to get a book and the second user's purpose is to get something to eat. The level or category varies from user to user but the key point is that every user has a clear purpose, an obvious desire, or a request that could be clearly expressed in words. In these cases, the user will make actions driven by the purpose until it is accomplished.

Figure 2.1 is an overview of a purposive-walk user model. At the first moment, a user comes up with a purpose which is to get something to eat. Then the user will search for something to eat and then visit a restaurant and get a meal. An important point is that the user was able to come up with a verbal purpose by him/her self and made an action based on the purpose that is to get something to eat.

2.1.2 Exploratory-walk

Exploratory-walk is a type of users which is not a designated user of existing navigation systems. Exploratory-walk users do not have a verbal purpose but has a desire to do "something". For example, exploratory-walk could

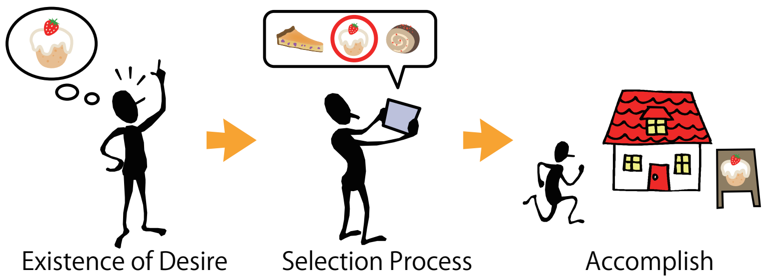

Figure 2.1: Overview of Purposive-walk Model

start when a user was suppose to meet up with a friend but it was cancelled suddenly after he/she arrived to the meeting point and lost his purpose of the following 3 hours but still wanted to do something rather than just going back to home. In this case, the exploratory-walk will start from the point that the user lost his purpose but has a motivation to do “something”.

The first step of the exploratory-walk will be to figure out what is “something” for the user. The user might be able to figure out by walking around the town and find something interesting or the user could visit a bookstore and thumb through a guidebook to see what is available around there. But these solutions are only available when the user knows the town, has courage to walk around a town new to the user, or was able to find a bookstore right in front of them. Because if the user was not able to find anything interesting in a meantime, this means that the user wasted a time and also if the place is new to the user it would be a pain to remember how to go back to the point the user started. In this situation, it is straight forward that it would be helpful if the user were able to get an information of possible candidates of “something” instantly without moving around and wasting the time.

Figure 2.2 shows the overview of the exploratory-walk in case when a user had three hours without a verbal purpose and wanted to do “something”. First, the user finds a furniture store that looks interesting to him. After looking around the furniture store, the user thinks “what to do next?” and then finds a coffee shop around the corner and visits and rest for a while.

The key point of the exploratory-walk is that the user does not have a verbal purpose at the beginning and throughout the activity. We discuss more deeply about exploratory-walk user model in Section 2.2

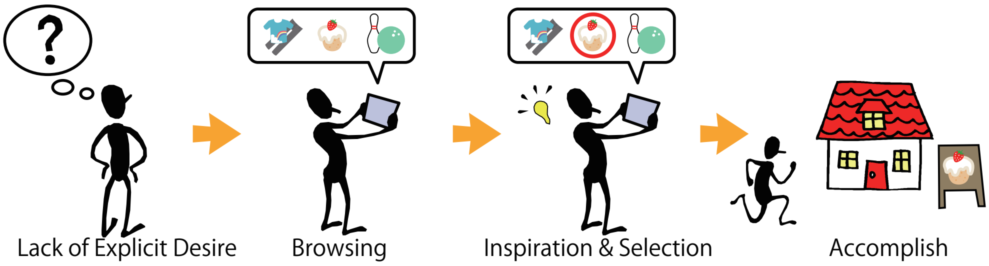

Figure 2.2: Overview of Exploratory-walk Model

2.2 Exploratory-walk User Model

As we defined previously, supporting an exploratory-walk user is our main interest and we discuss more deeply about characteristics of exploratory-walk user model. We would describe more precisely about the exploratory-walk user model by giving five key characteristics that represent the model.

- Unexpected Empty Time-slot
  Users have either rigid or loose amount of time unexpectedly.

- Transportation method is only walk
  Exploratory-walk users will spend time in a walk distance.

- No Goal or Obvious Purpose
  Users do not have goal or purpose that could be clearly expressed in words.

- Unstable Motivation
  Since users do not have a goal and a purpose, a motivation of an activity could change easily with any stimulation such as a discovery of an interesting shop or a good smell from a coffee shop.

- Variety of Situation
  Users' situation is unclear since there are a variety of possible situation. For example, a user might be in group or single, and might be hungry or full. And the user does not completely recognize their situation by their own.

These five characteristics could be divided into two categories: restriction and ambiguity of users. In Section 2.2.1 and Section 2.2.2, we explain about each category and characteristic that belongs.

2.2.1 Restrictions

Restrictions are factors that limit the user's activity. For example, walking will limit the distance that they could move at one time and also the region that they could move around. And of course a time-slot will limit the number and categories of places that a user could visit during their activity.

Empty Time-slot

As we discussed in Section 2.1.2, exploratory-walk starts when a user has an empty time-slot to spend and does not have any purpose. Therefore, the user will have empty time-slot for sure.

If the user has an event following right after the time-slot such as a next appointment with another friend or a train to go home, the user will have "rigid" amount of time which means the time to end exploratory-walk is fixed and unchangeable. On the other hand, if the user does not have any specific time limit, the user could choose any time to finish exploratory-walk and go back home or work on a purpose that has been forgotten.

It is clear that a user has an empty time-slot that occurred suddenly, and there is a possibility of both rigid time-slot and loose time-slot. And the user recognizes whether it is rigid of loose about his/her time-slot. And in some cases, the user might have unlimited amount of time for exploratory-walk and the system must be able to handle those cases.

Walking

Walking around for a long time is a tough work. If it is not an exercise, it often becomes as a cause of tiredness. Therefore, users cannot walk more than their capability depending on their profile and condition such as age and physical condition.

Walking will limit the user in the aspect of distance and speed of transportation. A typical speed of walking is defined as 3 - 5 km/h depending on source and in this thesis, we define 3 km/h as a speed of walking since the user might be wandering around rather than walking straight ahead to a goal. A length of time that the user could walk at once is limited too. The limit of distance varies from person to person, but for the most users, 30 minutes will be enough for a walking without stopping by and this will be 1.5 km in distance. Especially for users with rigid time-slot, distance and speed will be important to make sure that they could transit to the next activity without any problems.

Also usually, managing time, location, and distance that they walked are difficult for users.

2.2.2 Ambiguity

Ambiguity is factors that represent the ambiguity and unstableness of the context and the state of an exploratory-walk user. For example, since the user does not have a verbal purpose that lasts until the end, the user's motivation of walking or visiting a spot will be unstable. Also, the user might be alone or also there might be a case that the user is in a group. There are three key aspects that represent the ambiguity of an exploratory-walk user.

No Goal or Obvious Purpose

The base characteristic of the user is that he/she does not have a goal or an obvious purpose. This will cause unstableness when the user is deciding where to go because the policy of selection is unclear even for the user. Also, since the user does not have any goal, it is difficult to assure that the choice that the user made or a time that the user spent was the best choice. But the important point is that the user is satisfied with the way the user spent.

Users having no goal or obvious purpose will cause unstableness to decision making and evaluation of the way they spent their time-slot.

Unstable Motivation

Since users do have neither a goal nor a purpose, a motivation of an activity could change easily with any stimulation such as discovery of an interesting shop or a good smell from a coffee shop. A user could once decide to grab a snack at a donut store, but change his/her mind at front of the convenience store and stop by there. The user's temporary goal or motivation could change easily during the way.

Variety of Situation

Users' situation is unclear since there are a variety of possible situations. For example, a user might be in group or single, and might be hungry or full. If the user recognized their ambient condition such as hungry, frustrated, or tired, the user will be able to find a verbal purpose based to those conditions and become a purposive-walk user. Therefore, the exploratory-walk user has a complex situation and does not completely recognize their situation by their own.

2.3 QoS: Quality of Satisfaction

As same as there are characteristics in the way users act during the time-slot, there must be policies to spend or create a useful and/or satisfactory time-slot. We define the quality of the time-slot that the user spent as QoS: Quality of Satisfaction. In this section, we explain what kind of factors constructs a QoS for exploratory-walk users.

QoS is a combination of various factors which will also become a part of satisfaction. The factor could be how much did they enjoy, how easy it was to make decision, how efficiently they used their time, and more. Since QoS is a subjective scoring policy of time, it is difficult to standardize and there are many types of approaches in other navigation systems and even other systems that must measure the level of satisfaction of the user.

First we divide satisfaction into two types: system oriented satisfaction and user oriented satisfaction. A system oriented satisfaction is a satisfaction that is created or affected strongly by an external factor which is not possible to achieve if the user was alone and did not have any resource with him/her. For example ease of selection could not be achieved by user since the ability of selecting or making decision could not change extremely by an internal factor. User oriented satisfaction are the opponent of the system oriented satisfaction, which was created from a users decision and action. For example, the satisfaction of discovering new spots that matches to the user's preference or satisfaction of matching on spot and the user's preference.

Each type of Satisfaction will be describe including examples in Section 2.3.1 and Section 2.3.2.

2.3.1 System Oriented Satisfaction

System oriented satisfaction are satisfaction that could be achieved with a support of external factors. There are four key aspects of system oriented satisfaction: completion of selection, value of choice, cost of selection, and ease of return. Each aspect will be explained at the followings.

S1: Completion of Selection

Since the exploratory user is not sure if they will be able to find an interesting spot to visit at a new place, completion of selecting a spot to visit is an important factor of a satisfaction. If the user does not want to walk around without any knowledge, the only help that the user could receive is a support from a navigation system. Therefore, this aspect belongs to a system oriented function.

Completion of selection will be evaluated in simple axis. Was the user able to find a spot to visit or did the user need to walk around without any idea.

S2: Value of Choice

Another important aspect is how good the choice that the user made was. For example, if the user figures out that there was a more interesting spot to visit on the way back to home, which would be a disappointment for the user. The user will be satisfied when the user felt that they have made the right choice and was able to enjoy the time.

The right choice could be supported by choosing a spot from a large number of candidates since the user knows that it was the best choice in a large number. But once the user find other interesting place, it will corrupt. Of course if the spot matched to the user's preference and condition, the choice would be evaluated as a good choice from the user.

S3: Cost of Selection

How difficult it was to make a decision or a selection is another aspect of satisfaction. If the user had a hard time or took a long time to make a decision, this could lower the level of the user's satisfaction. And if the user was able to make a decision in a second and spent most of their time enjoying, it will be evaluated as successful.

Cost of selection could only be changed by external supports such as advices, recommendations, and suggestions. Asking to a person who looks familiar to current location might help getting a good advice, but asking to a person is high cost action for some users and also the quality depends on a person who asked the user.

S4: Ease of Return

Since a user will need to go back to his/her starting position or a mandatory location at the end of the time-slot. Usually the user must memorize the way back or the way to the end location but this is a stressful factor. For most of the users especially who are not good with recognizing location of them.

If the user were able to forget about time management and location recognition, the user could focus on which to decide and how to enjoy the spot to visit and this should give a good influence to the value of choice.

2.3.2 User Oriented Satisfaction

Unlike system oriented satisfactions, user oriented satisfactions are satisfactions that are produced inside the user such as feeling of freedom or supported, feeling of accomplishment for selecting the spot to visit. There are three key aspects of user oriented satisfaction: freedom of activity, fun of surprise, and satisfaction of selection.

U1: Freedom of Activity

Feeling free while an exploratory-walk is an important factor for a user. Since the user has an unstable motivation and condition, a user's decision will change rapidly. When the user feels trapped or restrained from changing their mind or condition, this will lower the satisfaction of the user. The user must be free to avoid or change the plan anytime. It will be better if the user was encouraged or supported when the user happened to change their mind.

U2: Fun of Surprise

Exploratory oriented user could also enjoy unexpected results such as discovering a new shop which the user never considered before. Purposive oriented users used to feel satisfied when the choice fitted to their preference since the purpose should never be disturbed. But exploratory-walk users have a potential and capability of accepting unexpected results since they do not have a purpose that would be an absolute axis of value. The user must have a chance to expand their region of interest.

U3: Satisfaction of Selecting

Feeling accomplished when finishing making decision is another factor of satisfaction. Usually, an amount of tasks to make decision which causes feeling of accomplishment became a cost and made the user tired. But for an exploratory-walk user, a correct amount of task to make decision has a possibility of increasing the level of satisfaction. For example, if the user were able to select a single spot to visit from one thousand candidates, the user might feel fulfilled because the selected spot seems to be the best out of thousand. But on the other hand this has a risk of making the user feel pressured or confused.

2.4 Summary

In this section, we defined and discussed what an exploratory-walk user model is and how do exploratory-walk users act and evaluate their activity. The exploratory-walk happens with a sudden occurrence of an empty time-slot without a explicit and/or verbal desire or purpose. And the exploratory-walk user has four main characteristics: Unexpected Empty Time-slot, No Goal or Obvious Purpose, Unstable Motivation, and Variety of Situation.
And the exploratory-walk user evaluates the QoS with seven axes: S1 Completion of Selection, S2 Value of Choice, S3 Cost of Selection, S4 Guarantee of Return, U1 Freedom of Activity, U2 Fun of Surprise, and U3 Satisfaction of Selecting.

The next chapter discusses how should a navigation system for the exploratory-walk user should look like.

Chapter 3

Exploratory-walk Navigation System

In this chapter, we clarify a coverage and requirements of an exploratory-walk navigation system by describing the position of exploratory-walk navigation system comparing to other types of navigation systems. First, we describe the definition of a navigation system in this thesis. Next, we explain the coverage and the purpose of the exploratory-walk navigation system focusing on differences with the others. Then, we describe requirements of the exploratory-walk navigation system based on characteristics of users and aspects of the QoS. In addition, we explain problems when the exploratory-walk user tries to use existing navigation systems.

3.1 Navigation Systems

In this section, we define the criteria of a navigation system by clarifying the user model and coverage of the navigation system. First, we define the user model and describe the flow of phases in the user model. Next, we explain the coverage of the navigation system by focusing on a transition support and management for the user.

3.1.1 Flow of User Activity

The starting point of a navigation system should be a user with a potential desire and a request. Then the user will recognize his/her desire, select an action, move, and fulfill the desire. As we discussed lightly in Chapter 1, in existing navigation systems, a user is supposed to have some verbal purposes when the user starts to use the navigation system. But we redefine the user model of a navigation system.

It is clear that person's action starts with some desires, requests, or a purposes. Users usually do not have an opportunity to focus on the process of recognizing their desire. Because in most cases, the desire, request, or purpose are obvious and verbal such as the case a user want eat a delicious case or a user want to buy a new wallet. But especially in the case of exploratory-walk described in Chapter 2, users have difficulty in recognizing their own desire, since they are in a situation that a primary purpose was crashed at that moment. Since users are having difficulty in the process of recognizing and verbalizing their desire and purpose, navigation should aim a user in that phase too.

Integrating a phase that the user has a potential desire and not verbal purpose will not change dramatically the user model of a navigation system, but its impact is huge against the system design of a navigation system. Before discussing the navigation system supporting recognition phase, we should clarify the user model and the flow.

A user model of a navigation system consists of five phases: potential desire and request, recognition of desire, select action, move, and accomplish. The user starts with having a potential desire and request. Then, the user will recognize his/her desire and verbalize it. Next, the user will select an action to fulfill his/her desire or accomplish his/her purpose. The user will need to move to accomplish the action and the user will be satisfied with the action. Finally, the user will accomplish and move for the next action.

The relationships and coordinations between each phase are described as an example shown in Figure 3.4. In Figure 3.4, a user starts with some

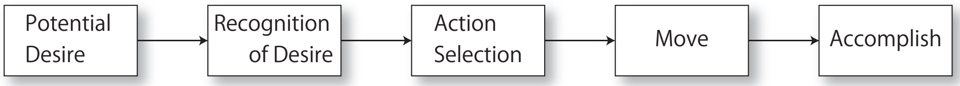

Figure 3.1: Flow of General User Model

ambiguous desire that he wants to do "something" but still cannot verbalize them. Then, the ad of a magazine that he discovers recalls that he wanted to get a new magazine released the day before. So, the user will choose a bookstore to visit, visit the bookstore, and then find and get the book.

We describe each phase at the following part of the section.

Potential Desire and Request Phase

First, a user starts with a potential desire and a request phase. A potential desire is a collection of ambiguous desires that are difficult to recognize and verbalize. A typical example is a situation that a person wants to "do something", but not sure what he/she wants to do. To recognize the desire, the user will need to think what does he/she wants to do within his/her mind or look around if there are anything that could bring an inspire. External or internal factors will drive the user to a recognition phase.

Recognition of Desire Phase

Recognition of desire phase starts with recognizing and verbalizing a user's desire and/or request. For example, when the user has not eaten anything for while, the user might recognize that he is hungry. The moment that the word "hungry" popped up in the user's mind is the recognition phase.

As the user's desire and request are still ambiguous even they are verbal, the abstraction level of desire is variable. For example, the user might stop thinking or categorizing his/her desire at the level of "hungry" or continue thinking until he/she narrows down the desire to the level that "want to eat delicious cookies".

As soon as the user recognizes and verbalizes the desire, the user will move on the next step which is selecting the next action.

Action Selection Phase

After clarifying the desire, a user will need to decide what to do to fulfill the desire. If the desire was to "eat dinner", the user would decide which restaurant to visit and what to order. If the desire was to "take a short rest", the user might visit a coffee shop or might visit an internet cafe. As shown in examples, the step to choose the solution of fulfilling the desire is a process to select a single answer from candidates in their mind or resources surrounding the user. Also it is important that there is a possibility of "re-thinking" which would change their mind to verbalize a different desire.

Move Phase

Moving is the next step of an action selection. After selecting what to do, the user will make a move for the action. It might be taking a walk to a coffee shop or get on a train to visit a friend. Also, working on homework could be a move too. Move could be used in both physical and logical context.

The cost and difficulty of move phase could vary by user. If the user has an outline of the map in his/her mind, it would be easy to move. But if the user is new to the place, it would be difficult to design the path of move.

Accomplish Phase

Finally, when a user reaches to the answer of fulfilling the solution, the user will enter to a phase of accomplishing the desire. Accomplishing the desire could be, enjoying a cup of coffee, watching movie, or finishing up the homework.

Throughout all phases the user is free to go back anytime he/she want in the logical flow map. In Section 3.1.2 we focus on what should navigation system do within this activity model.

3.1.2 Coverage of Navigation System

We define a navigation system as a system that supports user from the potential desire phase to the accomplish phase as shown in Figure 3.2. The support could be done in both physical and logical aspect. Four main functions of navigation system would be supporting desire recognition, solution discovery, transportation, and preparation. Also there are additional functions that could support the user throughout the whole phases. Four key aspects of the support are location management, time management, financial management, and history management. We discuss deeply for each component at the following.

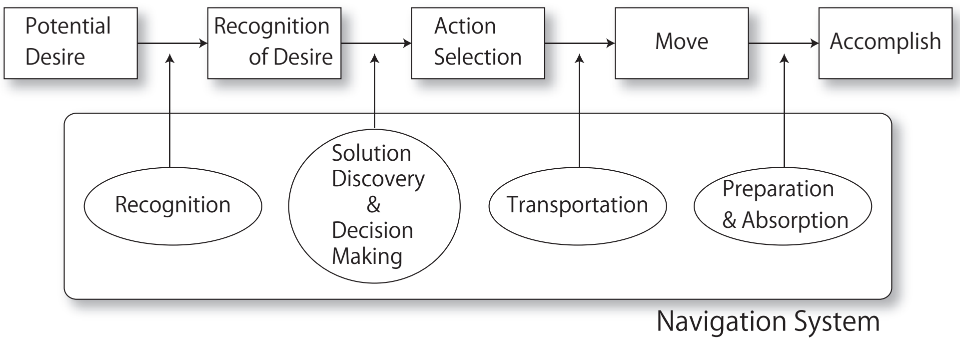

Figure 3.2: Coverage of Navigation System

3.1.3 Phase Transition Support

Supporting a user transiting from phase to phase is an important function of a navigation system. User might have a difficulty between each phase and also some external resources and advices could help users to discover a choice that would not come up if the user was alone.

We explain each case of support at the following.

Recognition of Desire

Helping a user to recognize or verbalize his/her desire is difficult for existing navigation systems. When a user is having difficulty with verbalizing the desire, a possible solution is to give many information and advices that could lead to an inspiration of the user. This will encourage user to recognize the potential desire and verbalize it.

Usually, this is done by information projection, suggestion, and advice.

Solution Discovery and Decision Making

Making decision on what and how to solve/fulfill the desire is another important transition in navigation system. Usually, the decision is made from the collection of choice which is in a user's mind. If the user has a verbal desire or a purpose on his/her own, it would be easy to select a choice since the user could make selection based on the requirements. For example, if the user wants to take a rest at a coffee shop, he could just decide which coffee shop to go. The system could purpose a group of choice that matches to the user's request. But if the user does not have any verbal desire or purpose, it would be difficult to make a selection, since there are no axes of evaluating spots. And the system would need to either predict some context of the user or just give a lot of choices or advices so that the user could get some inspiration.

Transportation

Difficulty of transportation often happens when a user do not know which direction he/she should move. The user could solve this problem by finding a public map on the street, or can get a map at a bookstore. So, the system could help user by showing a map or giving guidance to the user.

Transportation could be better if the path was efficient in an aspect of cost and also the amount of fun that the user had going through the path.

Preparation and Absorption

Preparation is important to fully enjoy the choice that a user made. For example, if the user did not know that a coffee shop next to the station is often crowded, and decided to take a rest at that coffee shop, it is clear that the user would not be satisfied because the shop was crowded.

The preparation could be done by a survey using an external resource such as guidebook and other user's opinions so that the user could assure the choice that he/she made is correct.

Also, it is important to fully absorb the joy of the selection. Visiting an amusement park is a typical example that shows preparation is important for absorption. If the user does not have enough information and missed the best attraction in the park, the user might still enjoyable with the park, but it is clear that there was a chance for more.

These are the key factors of transition support in the action model.

3.1.4 Management Support

Another aspect that the user could be supported is management. In addition to transiting the phases, users need to manage their location, time, and money to make correct decision and return to a correct place and time to move to the next schedule. But managing time and location usually interferes users to concentrate enjoying the spot they visit. Therefore, it is important to support management of location, time, and financial factors so that the user could concentrate enjoying and feel safe.

Another factor is managing the history. Usually, users manage a history on their memory or a notepad to memorize where they have visited and how they liked it. This becomes important information when they want to visit again or avoid visiting again and also when they want to look for a related or similar place. But unfortunately, memory of the user does not guarantee the length of memory and writing into a notepad could painful for some users. Managing a history could also be a key factor for raising the QoS and should be supported by the system.

Each factor will be described in the following.

Location Management

Location management is basically a function that could notify the current location of the user and how to get to a mandatory point based on map. The user should be able to access the map or get some other notion of location so that the user will not need to worry about how to get there and when to come back.

Time Management

Time management is a function that manages time-slot that the user has. The system should manage when the user needs to go back or how long will it take to visit a mandatory place. The user will not need to worry about time during the trip and concentrate on enjoying.

Financial Management

Financial management is a function to manage the money that the user has and/or how much will the selection cost for the user. Users are aware of how much it will cost because it affects whether the user could visit there or not. But since how much the user spends is flexible and the user could make a choice, managing the user's financial state is not that important and managing the cost of each spot should managed in acceptable style. For example, the system could manage an average amount of money that was spent for each candidate and tell the user so that the information could be an axis for making decision.

History Management

History management is a function that manages the history of activity of the user. For example, managing where the user visited in the past could help the user to avoid unexpected redundant visit. Also, it could be used for searching relative choice or similar choice from the history, which is often called preference based recommendation.

To fully utilize these functions, it is important that the navigation system concerns all of the factors that were described above.

3.1.5 Existing Services

As we have described in Section 2, an exploratory-walk model user cannot be satisfied by using existing services. Existing services do not expect exploratory-walk model and the exploratory-walk model user will face various problems when he/she attempts to use them. In this section, we describe existing services briefly and then explain what will be a problem when the exploratory-walk model user tries to use them. Figure 3.3 shows the position of each services and description will be described in followings.

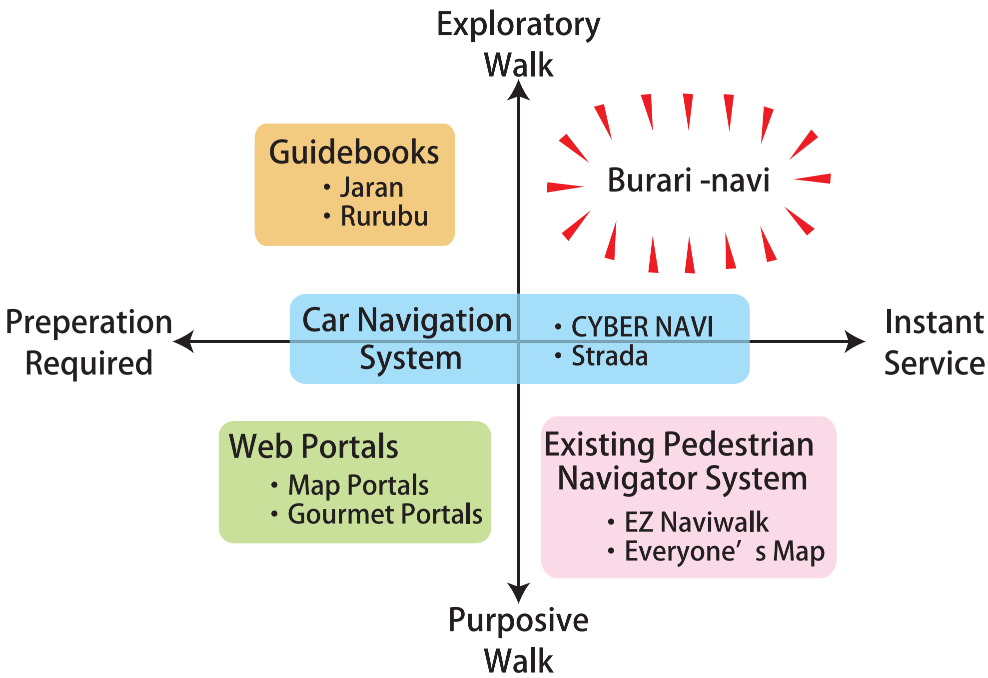

Figure 3.3: Mapping Existing Services

Car Navigation System

Car navigation system is the most popular navigation system in current market. Car navigation system used to support only the move phase in activity, but currently, the advance of technology has pushed them to expand the coverage area to plan phase.

CYBER NAVI and Strada are typical examples of advanced car navigation system that supports plan phase with new functions such as planning tool and spot search on it or by collaborating with web services.

- CYBER NAVI
  CYBER NAVI [8] is a car navigation system produced by Pioneer which has a big share in Japan. The strength of the recent CYBER NAVI systems are internet sync function that enables user to download new spot information from the database and even information on blogs related to spots. The connectivity to the internet will cover the weak point of the car navigation that it cannot update information easily.

- Strada
  Strada [9] is another car navigation system that has advanced technology and ideas built in. Strada enables users to add area information and trip plans into the navigation system by transferring the data from the Internet via flash memory cards. This allows users to find an interesting spots on internet and expand their choices. However, there is no function for downloading data while driving or also the purpose of the system is still to build a plan rather than just walking or driving around.

Pedestrian Navigation System

Another big category that is increasing its share rapidly is a pedestrian navigation system which is commonly deployed on mobile devices such as cell phones.

- EZ Navi Walk

  EZ Navi Walk [3] is the first system that spread widely in Japan and still keeping its position. EZ Navi Walk provides route guidance for pedestrian using cell phone as an interface. Before this service was deployed, guiding pedestrian using GPS and mobile device was difficult because GPS was still not popular and also it was not accurate enough to locate the position of the user. Since this service was deployed by a cell phone carrier, GPS began to be included on a mobile device and the number of users is growing rapidly. However, EZ Navi Walk only focuses on guidance and still does not have functions designed to support exploratory walk, such as a search based on time.

- NAVITIME

  NAVITIME [4] is a system that provides most rich functions on mobile devices. NAVITIME is a total navigation service that allows a user to find a spot and visit there with a route guide. NAVITIME has a strong algorithm that could calculate the best route combining many transportation methods and also has a support for driving. Even NAVITIME is strong with route guidance, still, it does not have an aspect of exploratory walk and requirements of inputs will be a big gap between the system and an exploratory walk user.

- Everyone's Map

  Everyone's Map [5] is a map-information-software for Playstation Portable released by Sony. It allows users to browse area information integrated into map and also has a function of route guide and information sharing via a website called "PetaMap". The way that the system has area information without search request and large number of choices is valuable, but the expression of choices and brows-ability is not good enough.

Map Portal

Map portals are popular navigation systems provided on web. Since most portals are built for typical web browsers on personal computers, they are powerful and have rich contents. But on the other hand, map portals are difficult to port them to mobile devices since devices do not have capability of rich contents within their small displays and other limited resources.

- Mapion
  Mapion [10] is one of the famous map portals that has variety of information and services but most of the services are provided on web which is difficult for a user to access via mobile device. Mapion also has a mobile service called "Mapion Mobile". This service enables users to look for spots that are close from the user. But since the spots are only sorted by distance and the user interface is not intuitive, it is still poor for exploratory walk users.

- Tripl
  Tripl [11] is a new type of a map-portal that enables users to build a trip plan on a virtual map on the web. Tripl is also not designed for mobile devices and the interface is not optimized for it. But the aspect that users collect their knowledge into one place is very important and tag based knowledge arrangement is advanced. Since Tripl is designed for trip planning it could not be applied for exploratory walk users, but still it has unique aspects and interesting interface.

- Sugoi Chizu
  Sugoi Chizu [12] is another map portal that allows users to pin a point on a map and attach name, image, comments, and tags on a location so that users could also retrieve many useful information based on keywords and themes. The way Sugoi Chizu arranges the data is useful but again, it does not support exploratory walk users since it requires input nor has some biased result set for users.

Gourmet Portal

Gourmet portal is a popular category of service that has been successful on both personal computers and mobile devices. But since it is a gourmet portal, a support for move phase is still weak and also the way it express sites and spots to visit is not efficient to pull out the potential desire of users.

- GOURMET NAVIGATOR
  GOURMET NAVIGATOR (GURUNAVI) [13] is the most popular gourmet portal on the web which also has mobile service as well. It provides set of restaurants that matches to a user's preference like category or atmosphere and other. It has an enormous number of restaurants stored in the database, but since the information is only about restaurants and the results are given based on query, it cannot give an efficient result to an exploratory walk user.

- Hotpepper
  Hotpepper [14] is another gourmet portal that is popular in Japan. Hotpepper also gives information of restaurants based on query such as an area and category. Recently Hotpepper has began expanding its coverage to beauty market such as hair salons and nail salons. The strength of this system is that they always include a coupon for products or dishes of the spot. This might help exploratory walk users to increase the motivation to visit, but some affection from users' comments or evaluation should be concerned too.

Guidebook

A Guidebook is not a computer aided system, but has been a strong assistant for people for a long time and still there a lot of fans of it. It has a critical weak point that it must be prepared or must find a bookstore to get it, but since it has some important strength, we use guidebook as an example as well.

- Jaran and Rurubu
  Jaran and Rurubu are both famous guidebook that have a variety of information for trip such as sight spots, restaurants, amusement spots, and hotels. Jaran and Rurubu have started online services on web few years ago and most of the contents are similar to guide books. But since the guidebooks must be purchased at a bookstore and online services are not designed for mobile devices and an exploratory walk user does not evaluate spots from the aspect of trip, these services will not fit.

3.2 Problems with Existing Services

Problems could be separated into three categories: entrance input trap, mismatching and small group of choices, and misfire of obvious interest. In this section, first we describe each category including some examples and then show that there are no existing services that will support exploratory-walk model by mapping existing services to the expected user characteristics which are described at Chapter 2.

3.2.1 Entrance Input Trap

Typical navigation system requires a user to input a requirements or restrictions in a form as keyword or a multiple selection such as genre and category. In these cases, an exploratory-walk user will have difficulty with inputting information to forms because they do not have any obvious interests of desire that are surfaced.

3.2.2 Mismatching and a Small Group of Choices

Even if the user was able to retrieve a set of candidates from the system, there is no guarantee that a policy of extracting candidates matches to the user's potential subject. For example, when the user chooses to use Gurunavi, the user will be unable to access any entertainment sights. Also if the user chose to use navitime and selected a genre "entertainment", user will need to go back to the top menu once to search other criteria.

3.2.3 Misfire of Obvious Interest

In the last phase of selecting a place to visit, the user will browse the candidates and look for interesting spots. But currently, some of the systems such as Google Maps [15] only provide text information at the top level. Text information is not the best approach to extract the potential subject from the user and some other intuitive and stimulating approach is desired.

3.3 Exploratory-walk Navigation System

In this section we discuss in details focusing on exploratory-walk users and their needs for navigation system. First, we clarify an exploratory-walk activity model which will be slightly different from a user model in general. Next, we discuss few aspects that must be covered and focused by exploratory-walk navigation system. And finally, we classify the requirements for exploratory-walk navigation system and problems with existing services.

3.3.1 Exploratory-walk Activity Model

A activity model of an exploratory-walk is slightly different from the general flow model described in Figure .

The overview of the flow model for exploratory-walk activity is shown in Figure 3.4. The main components are same as the general flow model but the flow usually cycles and the directions could change, reverse, and restart easily based on the characteristics described in Chapter 2.

We discuss deeply on the coverage of the navigation system and characteristics to be focused in Section 3.3.2.

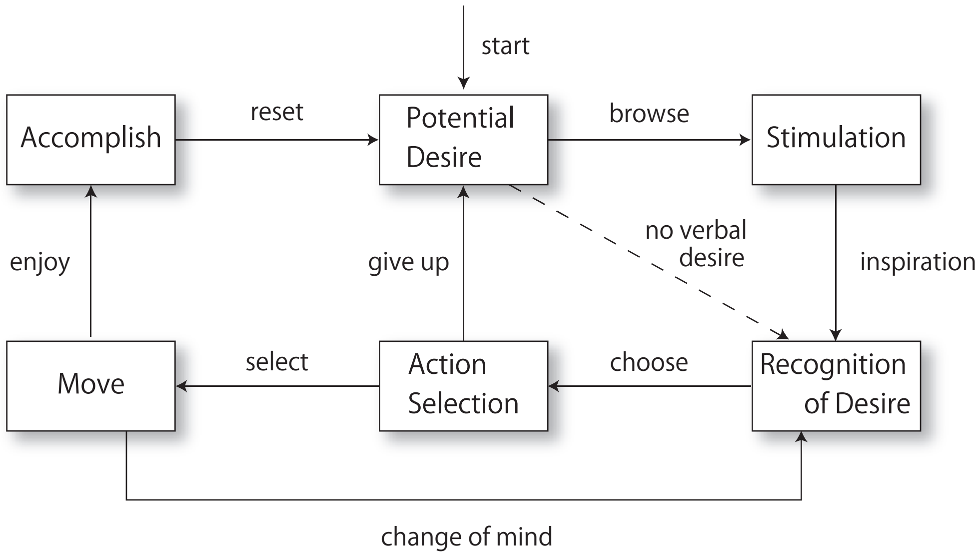

Figure 3.4: Flow of Exploratory-walk Model

3.3.2 Coverage and Focus

As the flow model of the exploratory-walk is different from other users, a navigation system designed for the exploratory-walk user should have a different focus on coverage since existing services do not fit to the exploratory-walk design as explained in Section 3.2.

In addition to the coverage of general navigation systems, the exploratory-walk navigation system should especially be sensitive on four subjects: difficulty of recognition and verbalization, cycle of flow, frequent reverse and restart, and limited time.

Each subjects are described in the following.

Difficulty of Recognition and Verbalization

Difficulty of recognition and verbalization is the most important characteristic and a problem that the exploratory-walk user owns. Lacking of verbal desire is a critical problem when a user tries to use existing navigation systems since the user will be required to input a keyword to throw a query to the system and get candidates that the user could choose. Even if the user was able to give some keywords to the system and get a collection of candidates, since the information is designed for purposive users, exploratory-walk users will have a difficulty choosing one of them.

Cycle of Flow

Cycle of flow is another point to focus when designing an exploratory-walk system. Since the user seems to have a cycle in flow model, the system needs to be ready for the next step all the time. Also the system might be able to get some characteristics of a cycle and use them as hints for searching.

Frequent Reverse and Restart

Frequent reverse and restart are caused by an ambiguous motivation of exploratory-walk users. Since the user does not have a verbal desire and a clear value function, there is always a chance for a user to find or encounter with a place that they like better and also to re-think of what they want to do. Therefore, the system should be aware about the unstableness of the flow too.

Limited Time

Limited time is another characteristic of an exploratory-walk which was not included in a purposive-walk. Since the exploratory-walk user does not have any verbal desire and stable purpose, a navigation system could not get any stable restrictions for searching candidature spots for the user. But, the limited time is one of a few restrictions that the user has and are able to express in a verbal form easily.

The navigation should not forget to use the factor of limited time as information and characteristics of the user.

3.4 Requirements of Exploratory-walk Navigation System

Now then, what are the requirements for exploratory-walk navigation system? As a conclusion of our discussion of exploratory-walk user and the navigation system designed for will be dropped into a form of requirements of exploratory-walk navigation system. Combining the characteristics of exploratory-walk described in Chapter 2 and coverage of the exploratory-walk navigation system in Section 3.3.2 we purpose a guideline of designing exploratory-walk navigation systems. There are three policies for designing an exploratory-walk navigation system: inspiring user interface, robust and flexible recommendation engine, and time and location management.

3.4.1 Inspiring User Interface

Inspiring user interface is the most important aspect of designing a navigation system for exploratory-user. And there are many aspect of designing the user interface such as efficiency, comfortableness, ease to use, and many other physical and emotional factors [16, 17] but we focus on providing a simple, consistent, and comfortable interaction for a user. Since the user does not have a verbalize-able desire or a request when start using the navigation system, the only way to move the user to the next step of the flow model is to give an inspiration by an action of the navigation system. It could be produced from a computer-user interaction or encouraged from a style of a user interface. For example, system could give a suggestion, a continuous question, or a collection of candidates to the user so that user could get some inspiration from the given information.

Inspiring user interface will solve the “Misfire of Obvious Interest” and “Mismatching and Small Group of Choices”.

No action on the system side cannot create constructive result and therefore it is required to the system that system makes and active movement again the user so that the user could get an inspiration from it.

3.4.2 Robust and Flexible System Flow Management

Because the user's behavior is unstable as discussed previously, the system must have a capability of absorbing the frequent changes of the user's location in a flow model. Even the user once decided to visit a coffee shop and then quitted and then decided to go to a bookstore, the system should react immediately and give the best service to the user depending on current status of the user.

This could be done by making the system's phase management quick and flexible and/or let one phase in the system to absorb the swing and perplexity in the user's attitude. For example, the system could be designed to switch the suggestion and navigation quickly and easily and/or the system could have a mechanism that the user could select, choose, and move at the same time.

Also, the interactive-ness of the system is important for exploratory-walk since the QoS will be affected by it.

And this should solve the "Entrance Input Trap" problem.

3.4.3 Time and Location Management

Time and location management is also a required ability for an exploratory-walk navigation system. Since users have a cyclic flow model and also has a possibility of changing mind, it is much difficult to manage their time and location by their own. Time management ability should help the user to manage the time that the user could spend and the time to go back. Location management ability should help the user to manage their current location and the way to visit a mandatory space. Both should collaborate since the functions provided by time management needs information produced by the location management ability. For example, time to go back would be calculated from the current location and the path to the location started.

Time and Location management are an important requirements for the exploratory-walk navigation system.

3.5 Summary

In this chapter, we clarified the coverage and requirements of an exploratory-walk navigation system by describing the position of the exploratory-walk navigation system comparing to other types of navigation systems.

The navigation system is a system that could support users from the aspect of transitioning and state management. The exploratory-walk navigation system should focus on difficulty of recognition, cycle of flow, frequent reverse and restart in flow, and limited time-slot.

We gave some examples of existing services and pointed out three problems: entrance inputs trap, mismatching and small group of choices, and misfire of obvious interest.

Finally, we defined three requirements of the exploratory-walk navigation system to solve problems: inspiring user interface, robust and flexible state management, and time management.

Chapter 4

Design of Burari-navi

In this chapter, we purpose an exploratory-walk navigation system called "Burari-navi". First, we describe the system from the user's point of view by explaining key features and a flow model of the system. Then, we focus on each key feature of Burari-navi and describe the characteristic of features and why those are the best for Burari-navi.

4.1 System Overview

Burari-navi is a mobile exploratory-walk navigation system that enables users to spend a satisfactory time without any verbal desire, request, or purpose. Usually, when a user did not have any verbal desire, request, or purpose it was difficult to get suggestions or advices of what would make the user satisfied. Even if the user were able to get some suggestions or any information from the system, since the user interface is not designed for exploratory-walk, it was difficult to get an inspiration from existing systems. In this section, we describe the overview of Burari-navi by using a scenario and explaining a flow of the system.

4.1.1 Scenario: Shibuya Station at 3PM

First, we explain the overview of Burari-navi based on a scenario described below. Then, we illustrate the flow of the system, and then focus on key

Figure 4.1: Use Image of Burari-navi

features at following sections.

Scenario:

Year 2010, one day at 3PM, a twenty years old, university student, Toshi was waiting for his friend Hide at Shibuya Station. Around 1:05, an e-mail from Hide tells that he could not come because he caught a cold and not in a good condition. Since they were planning to build a plan of the day after they meet, Toshi did not have any plans for the next one hour but wanted to do something because it took an hour to come Shibuya from his home.

So, Toshi takes out his cell phone and launches Burari-navi application. Burari-navi asks Toshi to input either he would like to come back or walk around and if there is a time limit. Since Toshi had an appointment with his dentist at 5PM, he decided to come back to the station after an hour from now.

Once he input the time-slot for an hour and select to return at the time and pushed the start button, Burari-navi shows Toshi a lot of choices to do in pictures aligned in grid.

When he browsed through the picture of a good looking coffee, an idea to have drink at a coffee shop came up in his mind. So he looked into a description of that coffee shop by zooming in to the picture and decided to head for there.

Also, he looked if there are many spots around the coffee shop by using the locator browser and also compared with other coffee shop by using the comparison browser.

Toshi selected that coffee shop and Burari-navi started to guide him to the coffee shop.

After spending 30 minutes at the coffee shop, he opened his cell phone and looked for next possible choices in the screen using Burari-navi's spot browser. He found a picture of a neat T-shirt and decided to visit the store which sells that T-shirt.

While he was looking through the store, Burari-navi alerted him that it is a time to go back on time he set at the beginning. Toshi walked back to the station being guided by Burari-navi and headed to the dentist.

4.1.2 System Flow

Burari-navi's state transition is visualized in Figure 4.2. First, the system will let the user input either to come back or walk around in a specific time-slot. Depending on current location, remaining time-slot, and an activity history if available, the system will show a collection of spots in a style of picture grid surface which is draggable by a finger on a touch-screen. Burari-navi requires touch-screen and 4 physical/virtual buttons as an input which could be easily satisfied by recent smart phones such as iPhone.

1. The user inputs a time-slot and a need of return to Burari-navi and press the start button.

2. Burari-navi will show a large number of choices to select within a grid style interface. The interface will automatically focus into a spot that is placed in center of the screen. The user can press-down the screen

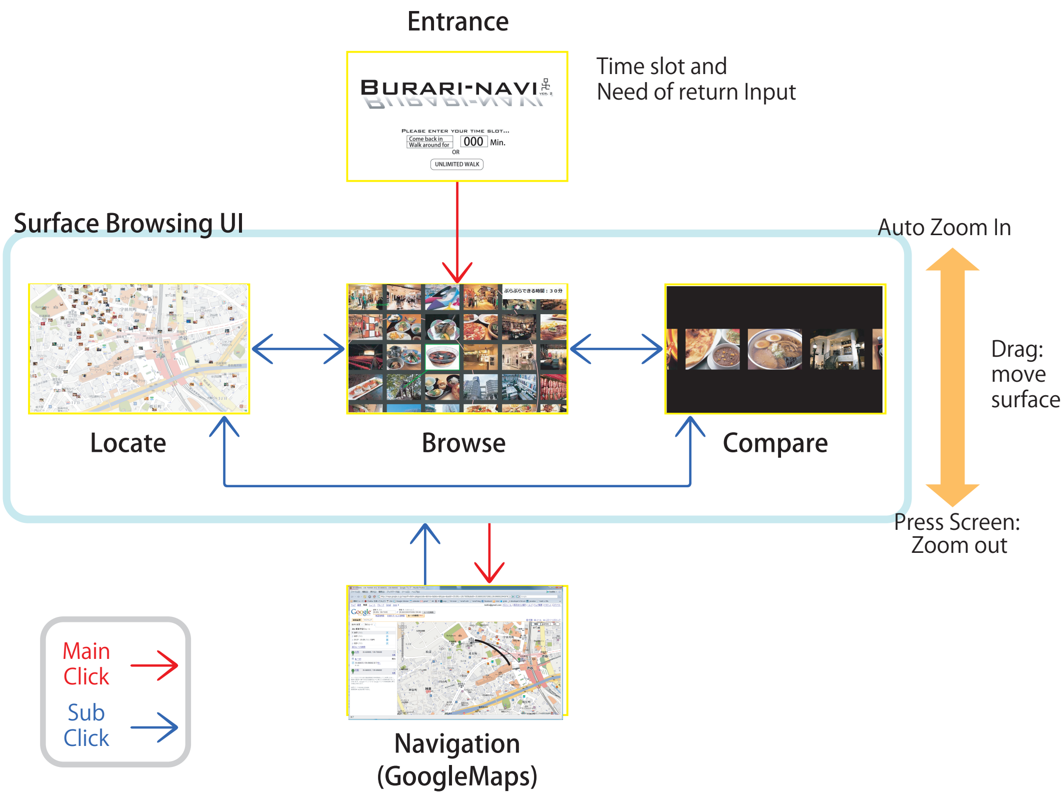

Figure 4.2: System Flow of Burari-navi

to zoom out and drag the screen to move the focus of the interface.

3. If the user wants to assure that the area around the focused spot is not too far and located in a place with a good atmosphere, the user could click the sub-button and shift the interface to the Area browser which shows spots mapped on a geographical map.

4. And if still the user are not sure that the selected spot is the best choice, the user could click the sub-button again and then a Comparison browser will appear. Comparison browser only shows similar spots to the spot which is currently focused by the user. And the user could compare between similar spots and make a decision.

5. The user could make decision in either all three browsing interface. And once they make a decision, the user will click the main button and transit to a navigation phase.

6. The user will be guided to the selected spot by the navigation interface.

We focus on each key feature from Section 4.2 through Section 4.6.

4.2 Time-slot-based Entrance

A first feature to focus on is a time-slot based entrance which only requires an input of time-slot and where to end the journey at the entrance of the system.

Most of the existing systems required a verbal word, category, or a key-phrase as the input at the entrance, but as mentioned, this is a difficult process for a user and an exploratory-walk system should support from that verbalization phase.

Therefore, Burari-navi will have only two inputs from the user at the entrance: time-slot and a need of return.

4.2.1 No Preference Request

The important aspect is that the entrance of Burari-navi does not require users to input verbal desire or preference which is not verbalized yet. This will lower the first step of using Burari-navi for an exploratory-walk because a verbal preference input is often difficult and stressful for the user.

On the other hand, no preference request will cause a problem of the system. Since there are no extra restrictions for filtering spots, it would be difficult to filter spots from a giant collection of spots. Burari-navi solves the problem by changing the purpose of the system from an accurate suggestion to wide variety of choices and freedom of selection. But of course, Burari-navi has a mechanism to avoid the risk of users being overwhelmed from a huge set of choice. Overwhelming avoidance mechanism is provided at zooming of the user interface and sorting of the recommendation engine.

4.2.2 Time-slot Management

By using the input of a time-slot by the user at the entrance, Burari-navi will manage the time for the user. Time management is an important factor for an exploratory-walk, because usually there are other schedules waiting for the user and it is difficult to manage the time to go back when walking around exploratory, and Burari-navi has a feature for it. Burari-navi would show the remaining time on the user interface anytime the user wants to check. Burari-navi will alert the user by an actuator such as a speaker and a vibrator when the user needs to move to arrive at the end location on time. The timing of an alert will be calculated using the location of the user and the distance from the end location.

4.3 Surface Browsing UI

Next key feature is the Surface Browsing User Interface which enables users to browse through a large amount of choices within the speed and scale that they like. In this section, we clarify the strength of Surface Browsing UI by comparing with other methods purposed in related works.

4.3.1 Flexible Brows-ability

Flexible brows-ability is the most important factor of Surface Browsing UI. In Surface Browsing UI, a user could browse through choices by dragging a virtual surface on the screen. Also the user could adjust the scale of the surface by them. This enables users to browse through choices within their capability efficiently.

4.3.2 Browsing, Locating, and Comparing Spots

The Surface Browsing UI of Burari-navi provides three types of surfaces to browse: thumb-through surface, area surface, and comparing surface. All three surfaces have the same interaction style and shares the spot that the user is interested in so that the user could shift each surface without confusion and comfortably.

- Thumb-through Surface
  Thumb-through surface has the highest brows-ability and encourages the user to thumb-through a large number of choices to meet a lot of choice and get some inspiration from the impression of choices. The choices are sorted in order of distance but concentrated into a grid style to increase the density of information.

- Area Surface
  Area surface enable users to figure out the atmosphere of the area that the current interested spot is located. It does not have high brows-ability but provides a sense of location and atmosphere of spots to the user. Spots are mapped on a graphical map so that the user could feel the distance, the direction, and if there are a lot of spots surrounding or not.

- Comparison Surface
  Comparison surface enable users to compare a spot that he/she is

interested in and other spots that has similarity or are in a same category. This supports users to make decision when users are interested in some characteristic of the spot but still not sure if he/she really wants to go. By comparing with similar spots, the user might be able to find better spot or assure that the spot originally focused is the best.

4.3.3 Avoidance of Overwhelming

Surface Browsing UI is a browsing interface which is designed for exploratory-walk users. Surface Browsing UI has both interactive-ness and less restriction input which will raise a QoS of an exploratory-walk user.

The difference from the other browsing interface is in the flexibility of controlling the speed and scale. Cover Flow on iPhone does not have enough brows-ability since it has only one dimensional space to browse through and Google Earth is to complicated to browse through a collection of choice rather than whole information on earth. InfoGlobe also has a three dimensional space as Google earth and it is complicated for the user to have interactive-ness. Gards has the most similar interface but does not have a flexibility of scale.

Therefore, the Surface Browsing UI should be chosen for Burari-navi.

4.3.4 Multi Level Information

Multi Level Information is a factor that supports the flexibility of the Surface Browsing UI. In Surface Browsing UI, each choice is shown differently to the user depending on current scale of the virtual surface. When the choice is fully displayed on the screen, the expression will change to descriptions rather than just a single picture because the scale of the surface could be translated into the strength of the user's interest.

In Surface Browsing UI, the expression of choice will change as shown in Figure 4.3. At the lowest level of zooming, the choice will be shown by a picture but as the choice gets zoomed in to a certain level, a new layer of information including category, name, and description will appear.

Figure 4.3: Example of Multi Level Information

4.4 Location-aware Gentle Recommendation

Gentle Recommendation is a second key feature of Burari-navi. To enable users to browse spots efficiently on Surface Browsing UI a function that enables users to view important choices earlier than others.

In Burari-navi a collection of choices will once be sorted in an order that is important for the user. And then it will be mapped on the virtual surface. Since the first spot in the group will be in the center of the interface, the mean time to be in a sight of the user will be shorter as the spot get high priority.

Existing services implements recommendation engine and sort the choices on the user interface. For example, in Groumet Navigator [13], the system will show a limited sized list of choice to the user which is sorted by cost, category, or other factor which is chosen by a user. By using a limited and sorted list, the system will lose a brows-ability. Therefore, a system must have a gentle recommendation mechanism that enables the user an efficient browsing and still keeps the brows-ability.

Also, an obvious sorting on a user interface interferes the freedom of selection for the user which is an important factor of a QoS of a user.

4.4.1 Ambient Sorting

As explained, a collection of choices will be sorted once and then mapped to the virtual surface in Burari-navi. This will bring important choices for the users to the center of the surface so that the user could browse those choices first. But to keep the freedom of selection of the user, Burari-navi will not select the important choices automatically or drop non important places.

4.4.2 Interactive Recommendation

The order of the choices will change dynamically depending on the context of the user. For example, if the user's remaining time-slot becomes less than an hour, the choice of a movie theater will be eliminated. And if the user has visited a coffee shop, when the user looks for the next choice, the system will lower the priority of other coffee shops since it is rare that the user would like to visit multiple coffee shops continuously.

Interactive recommendation will enable efficient browsing and pleasure of selection at the same time which is both a factor of the QoS of the user.

4.4.3 Scoring Policy

The collection of choices will be updated when the user moves or an unreachable spot has been discovered from the manager depending on remaining time-slot and required time of the spot. The policy of filtering and sorting are as follows.

1. Reachable or Not: the system will drop the choice that are not reachable or cannot enjoy because the remaining time are smaller than the required time of spot.

2. Link Collaborative Filtering: the system will raise the score of the choice that has a strong link (many users have visited next) with current location.

3. Similarity Scoring: the system will raise the score that has similarity in description of the spot.

4. Redundancy Avoidance: the system will avoid redundant recommendation by lowering the score of redundant choice such as a restaurant right after visiting a restaurant.

The policy and the mechanism will be discussed deeply in Chapter 5.

4.5 Action-based Preference Manager

Action-based Preference Manager is a module of Burari-navi that predicts the preference of the user without asking directly to the user. Action-based preference manager will focus on a log of interactions made between the user and the system while the user is browsing through the choices, moving to the spot, and spending time at a spot.

4.5.1 No Stress Preference Fetching

Usually, navigation systems tried to get the preference of the user by profiling based on Q&A or a history of active inputs from the user. Q&A is a stressful process for a user because even if the user knows that will improve the quality of choices, but still it is better if the choice would be shown without answering to questions. Especially if questions are related to the desire or a request, it would be difficult for exploratory-walk users to do.

Burari-navi will not give such questions that are difficult for users. Instead of asking questions, the system will observe the action of users from the background and use the recorded logs to predict the user's preference.

For example, the record of what the user have chosen, how fast have the user walked, how long have the user stayed could be an hint for what the user likes, how fast can the user move, and what kind of activity pattern does the user have.

4.5.2 Collaborative Information

Actually, few kinds of records do not work efficiently if the record is stored only for a single user. The power of the records will increase when it gets together with other user's record. For example, if there are many users visiting to a T-shirt store "A" after visiting T-shirt store "B". The system could assume that the user who visits A might like B too, and it would be important factor when sorting the choices. Before the Internet became as popular as mobile devices also have connectivity, it was really difficult to collect user activities and use them dynamically for suggestions and recommendations because there were no easy methods for collecting all users' activity record. But now, almost all mobile devices such as cell phone have connectivity to the Internet and this made it really easy for the system to collect and use a huge number of users activity and this lower the cost of predicting user's preference.

By using all users' activity records, Burari-navi will be able to sort choices into the order that might be relevant or important for the user without asking questions to the user.

4.6 Guidance Manager

In this section, we describe the third key feature, the Guidance manager. Guidance manager of Burari-navi provides intuitive guidance to a user from two aspects: user-centric guidance and low resolution guidance.

4.6.1 User-centric Guidance

Maps must be designed within a user-centric policy [18]. This is an important policy when a developer designs a system. The guidance interface should be designed from the user's point of view too.

When the user is following the guidance on a mobile device, there are three steps until the user decides the direction to move.

1. Connecting the Location of the user and the map

2. Adjusting the Angle of the user and the map
3. Recognizing the scale of the user and map

Recently, few mobile navigation systems such as EZ Navi Walk [3] has been reducing the cost of each step by optimizing the guidance interface for users. The location of the user will always be center in the map and the direction "up" in the map will be same as the direction that the user is walking. The scale could be changeable so that the user will feel comfortable and also since the map is moving with the user, it will become easier to recognize the speed of walking in the map too.

4.6.2 Low Resolution Guidance

In the moving phase of exploratory-walk, users will need a support only for reaching to the spot and the user interface usually does not have high resolution display.

Burari-navi provides a simple and user-centered guidance manager that is optimized to guidance and no other functions. If the user changed their mind and want to look for other choices, the system could step back to the selection phase with one click from the user.

4.7 Summary

In this chapter, we described the overview of Burari-navi mainly from the user's point of view, based on the scenario and the system flow. We also clarified strengths of key features comparing with other approaches that could be applied. No preference input allows users to browse through choices without any explicit input of desire. Surface Browsing UI enables users to browse choices in three surfaces: thumb-through surface with high brows-ability, area surface to grasp the atmosphere, and comparison surface to compare with relevant choices. All surfaces use the same touch-screen interaction to enhance intuitiveness and brows-ability of the user interface. Then the simple user-centric navigation interface guides the user to the selected choice. In the next chapter, we describe the implementation of the Burari-navi system.

Chapter 5

Implementation of Burari-navi

In this chapter, we explain how Burari-navi is designed and implemented from the system's perspective. First, we illustrate the system from both hardware and software point of view. Then, we focus on each component of the Burari-navi software and describe the mechanism and the current implementation.

5.1 Hardware Components

Burari-navi consists of four hardware components: Central mobile device, Location sensor, Spot & activity database and Map service. The relationships between each component are described in Figure 5.1. Location sensor could be either built in to the central mobile device or attached externally using wireless interface such as Bluetooth. Central mobile device communicates with spot and activity database and map service via the Internet.Each component's function will be descibed below.

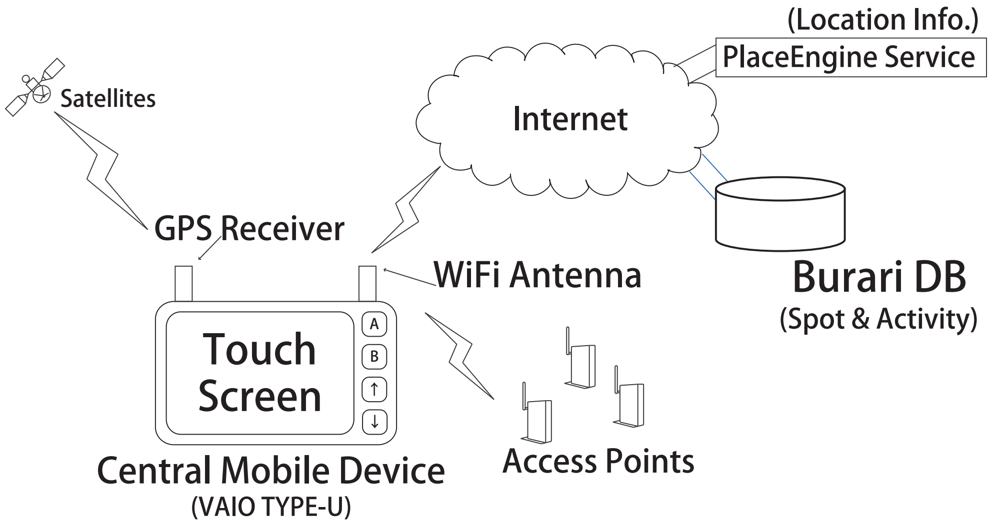

Figure 5.1: Hardware Components of Burari-navi

5.1.1 Central Mobile Device

Central mobile device is the heart of Burari-navi system. It should have an ability to present a user interface to a user, communicate with other components, and manage the state of the system and the user. Also being portable is a necessary factor since the service should be accessible any time requested.

We have implemented the Burari-navi system on Vaio Type-U which is an portable computer sold from Sony which runs Windows XP on it and has 2.2 inch touch-screen, multiple wireless interface, and acceptable amount of resource. Specifications of Vaio Type-U are described in Table 5.1 and the appearance is shown in Figure 5.2.

5.1.2 Location Sensor

Location Sensor is a device to detect the current location of the user. It could be either a portable sensor device or a sensor embedded in an environment. Since the portable sensor device such as GPS receiver is popular and much cost efficient, we chose to use GPS receiver as a location sensor.

Table 5.1: Specifications of Vaio Type-U

Item	Value
Size	167mm (Width) x 108mm (Height) x 26.4mm (Depth)
Weight	550g (approx.)
Display Size	4.5”
Display Resolution	1024 x 600
Input	Touch-Screen, Button x 2, TrackPoint, Keyboard
Output	Sound(Speaker)
Processor	Intel Core 2 Solo (1.2 GHz)
Memory	512MB
Operating System	Windows XP SP2
Wireless Communication	Bluetooth, Wifi

Figure 5.2: Vaio Type-U

And in addition to GPS, we have decided to use PlaceEngine [19] to get more accurate and instant location detection at indoor. PlaceEngine is a location detection engine that predicts the location based on the signal strength of beacons of Wifi access points that could be received.

Vaio Type-U does not have embedded GPS so we used Nokia's GPS module called LD-3W which could communicate with Vaio Type-U via Bluetooth. The GPS module sends out the location in NMEA0183 format which consists of various information related to satellites and location including Longitude, and Latitude in WGS-84 metrics which we use for Burari-navi. Also the strength of LD-3W is that the position pinning function is turned off which means that it could tell us a small movement of the user using raw data with a risk of low accuracy comparing to automatically corrected value. Figure 5.3 shows the appearance of LD-3W.

Figure 5.3: GPS Receiver(Nokia, LD-3W)

5.1.3 Spot and Activity Database

Spot and activity database stores information of spots and all users' activity record. If the system has a large number of users it should be an enterprise server, but since Burari-navi is built for experiment and has only few users, it is build on an off-the-shelf 1.2 GHz CPU computer which has Postgresql server running on Linux in it.

5.1.4 Map Service

Map service is a component that provides geographical information such as visual maps and route information. It could be built by our own but since there have been an increasing number of open services on the Internet, we have decided to use Google Maps API [15] which is an open API to use a map service provided by Google.

5.2 Software Components

Software components of Burari-navi are complicated comparing to the hardware. There are four large software components: Burari UI Module, Map service, Burari Spot Manager, Burari User Manager, and Databases. In this section, we clarify the position and overview of each component. And precise information and mechanisms inside each component will be discussed in the following sections. The overview of the system is illustrated in Figure 5.4.

- Burari UI Module

  The user interface module interacts with the user by receiving input from user and showing a collection of list to the user.

- Map Service

  The map service will be requested to show a map and route on display depending on the state of the system.

- Burari Spot Manager

  Burari spot manager will manage the spots that would be shown as a choice to the user. It will receive state updates from the Burari user manager and update the spots by throwing a query to the database, evaluating the spots, and sorting them.

- Burari User Manager

  Burari user manager manages the state of the user by communicating

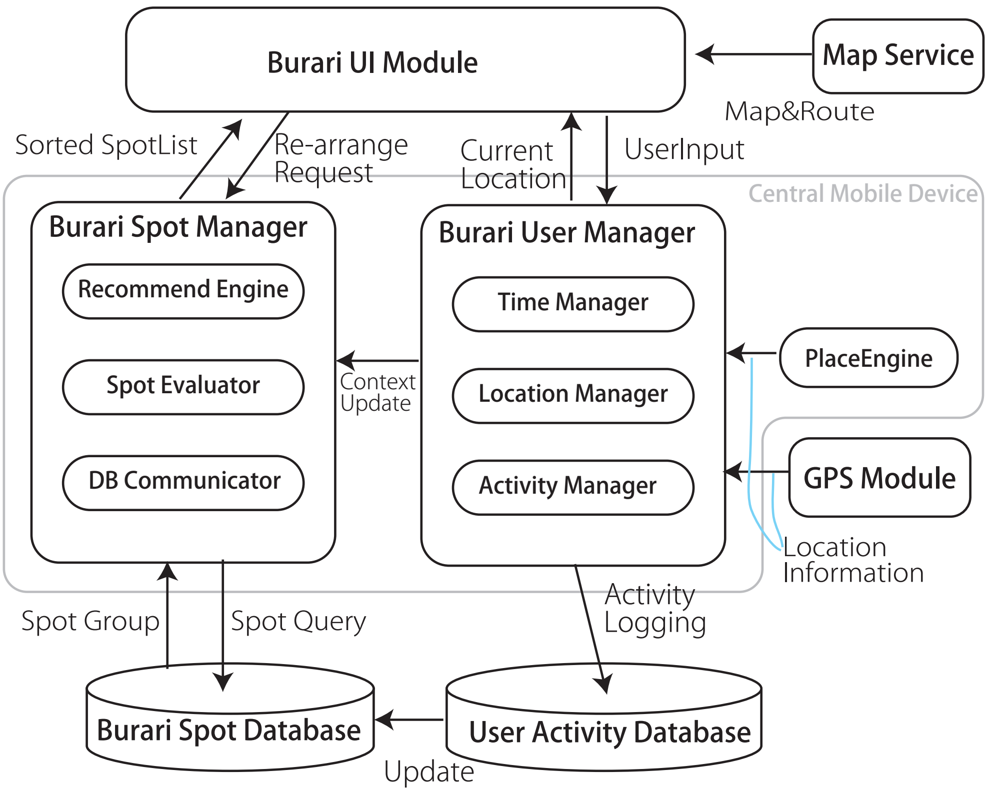

Figure 5.4: Software Components (Over View)

with the user interface and location modules. It also records log to the user activity database.

- Databases
  Stores and data such as spot information and user activity history for all users. Databases will return data that matches to the query and the data will be updated by the User activity manager in Burari user manager.

Following part of this chapter will be focusing on and describing each software components of Burari-navi.

5.3 Burari User Interface Module

User interface module provides three types of GUI to a user depending on his/her situation.

First, the user will browse spots located in the sorrounding area using Thumb through browser and find a spot to visit. Then, the user could get spot information mapped on a visual map so that the user can predict the atmosphere of the area from the density of spots and types of spots. Also, the user could compare with similar spots to assure the quality of the spot using Comparison Browser. After the user selected the spot to visit, Burari-navi will pass the information to a navigation system so that the user could reach to that spot without any problems.

5.3.1 Picture Card Expression

In both Area browser and Thumb through browser, each spots will be expressed as a Picture card. Picture card in Burari-navi is a rectangle card floating in a virtual space which has a thumbnail picture on the front and further information such as category, name, comments, and phone number on the back. Picture card expression enables user to get the impression of the spot immediately from the thumbnail picture and further information via description on the back.

Figure 5.5: Two Level of Picture Card Expression

5.3.2 Area Browser

Area browser is a user interface that enables user to find an area that seems to be interesting by mapping picture cards on a two dimensional surface according to a geographical location of the spot. Area browser will map spots passed from the spot filtering module on two dimensional surface using only the front side of the picture card. The center of the surface will be current location of the user and circles will be drawn radially for each five hundred meters to enable user to get a sense of a distance from the spot without interfering the impression and the information of the spot.

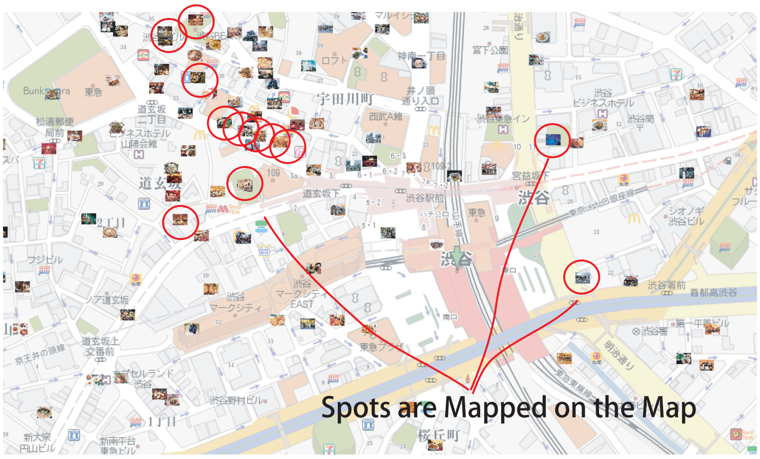

Figure 5.6: Area Browser

5.3.3 Thumb Through Browser

By using Thumb through browser, a user will select a spot to visit. Thumb through browser is a user interface that allows a user to browse a number of spots in a speed and granularity that the user desires like thumb through-ing a book. In Thumb through browser, spots will be arranged as a grid on a two dimensional surface in virtual space. The user will move a view point 3 dimensionally in the virtual space by dragging a finger on the screen so that the user could browse through the spots in a scale and speed that he/she likes. Spots located within one kilo-meter from a point that was selected in Area browser will be shown on Thumb through browser.

Two Dimensional Mapping

Thumb through browser uses two dimensional grid space to map the spots because it has a good balance of brows-ability, readability, and flexibility.

Three-dimensional space has a large capability of expression and system could show spots mapped on a globe, tube, and any other space. If the way the spot is mapped has a critical meaning or a significant impact to the users it might be discussable to choose three-dimensional space. But on the other hand, complex space will reduce the readability of information since the user will be required to recognize more complex information and browsing three-dimensional space often requires complex operation.

One dimensional space such as a line could be another choice. For example, cover view of iPhone by Apple Inc. [20] enabled users to browse through a line of a cover jacket of CD rather than text information to increase the brows-ability and the readability and intuitiveness at the same time. This method is effective but still lacks a capability of expression than two-dimensional space. Also it does not have zoom-in and zoom-out function which could be important for browsing picture of candidate spots rather than just an impression of a cover jacket. Another critical reason of not choosing one-dimensional space is that this method especially works well only with wide screen.

Therefore, with a good balance of brows-ability, readability, and capability of expression we chose two-dimensional space to map spots. And because a grid layout will increase the brows-ability clearly and easily, we arranged spots into the grid layout which has an order in the direction of spiral from the center.

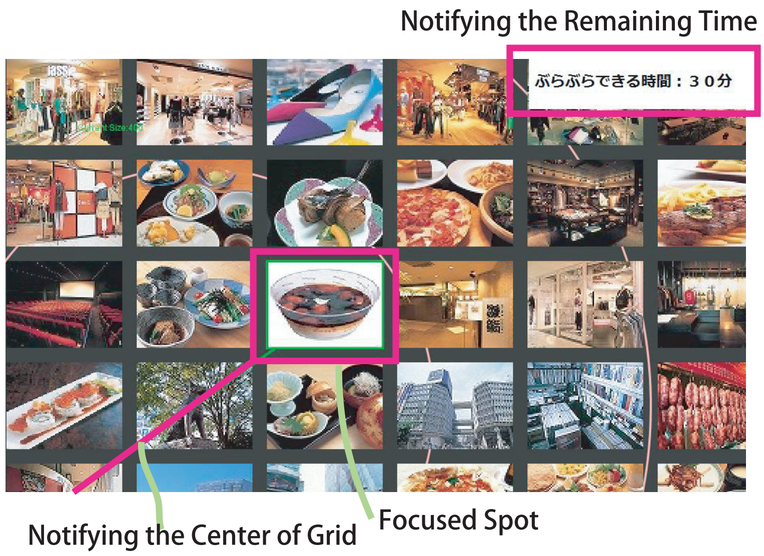

Figure 5.7: Thumb Through Browser

5.3.4 Comparison Browser

In addition to Area Browser and Thumb-through Browser, we have implemented a Comparison Browser which enables users to browse through choices that are similar to the focused choice. For example, if the user found a noodle store that looks good and is interested in it. If the user wants to compare with other noodle shop and similar shops, that is the moment for the Comparison Browser.

Comparison browser will show similar spots of focused choice which will be the noodle store. Purposed choices by the similarity browser could be Udon store, Spaghetti store, or other noodle restaurants that has similarity in category and tags.

Figure 5.8 is a screenshot of the Comparison Browser. The focused item will be shown at the upper center of the screen and similar spots will be arranged in line so that the user could browse through the spots within a same operation policy.

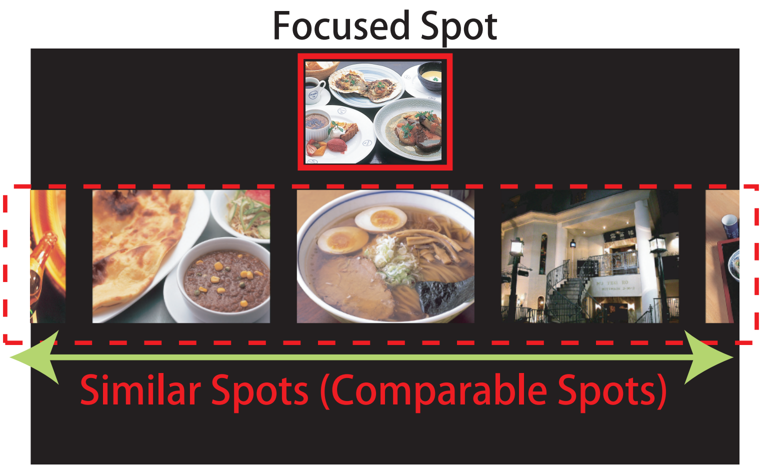

Figure 5.8: Comparison Browser

5.4 Burari User Manager

Burari user manager manages the state of the user. The architecture inside the Burari user manager is shown in Figure 5.9. A mechanism of each task will be described in following subsections of this chapter. Update will be done periodically by the time manager and the location manager. And also update could be done instantly by the user activity manager when a user have selected a spot. All information available in the Burari user manager will be sent together to the Burari spot manager.

Time manager receives the time-slot input from the user and keep telling the current time and the remaining time via updates. Location manager first memorizes end-location input from the Burari UI Module and then receives updates from GPS and PlaceEngine and tell the Burari spot manager the end location, current location and the average speed of walk. Activity

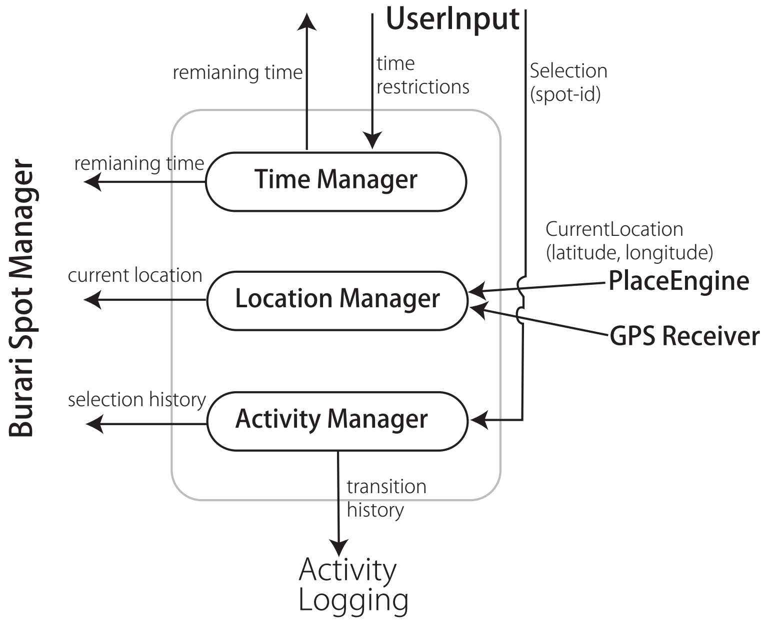

Figure 5.9: Architecture of User Manager

manager receives each selected spot from the Burari UI Module and record them. Recorded activities will be passed to the Spot Manager in the form of "spots selected today", "spots selected from the first use", and "currently selected spot". Also the selection of spot will be recorded to the User activity database to enable users to share the knowledge.

5.4.1 User Time Management

Time management is done simply by using the system clock. After the system is launched by the user, time manager will receive time-slot information in the format of minutes. The values could be set freely in the unit of 5 minutes from 15 minutes to 5 hours or unlimited.

Every 5 minutes, the time manager will send an update to the spot manager with the information of current time and remaining time-slot.

5.4.2 Hybrid Location Management

The User location manager uses GPS and PlaceEngine [19] to manage the user's location because the GPS would not be able to detect location at indoors. First, the return location will be memorized if needed, by recording the first detected location. The User location manager will send location updates of end location and current location within a format of latitude and longtitude.

- PlaceEngine
  PlaceEngine is a location-detecting system that is purposed by Rekimoto et al. at Sony. PlaceEngine predicts the user's location by measuring the beacon signal strength of the surrounding Wifi access points and referring the location of each access point recorded in the database. This method is useful when the GPS is not accurate such as walking indoors and between the tall buildings. Another strength is an 3 seconds cold boot, which takes about 3 minutes for GPS.

The User location manager will access to PlaceEngine via PlaceEngine API and fetch location information as main source of location because the system would be often used in urban area and the coverage of PlaceEngine should be supported. Since the PlaceEngine uses beacons from access points, there is also a chance of receiving location within a unit of spot rather than longitude and latitude. When the PlaceEngine could not detect any access points, GPS Module will be called as a backup unit and will return a location. Figure 5.10 demonstrates the accuracy of PlaceEngine in urban area by comparing paths created by each system and the actual path.

5.4.3 User Activity Manager

The User activity manager will manage the history of spot selection done by the user and send the collection of spots to Burari spot manager as a resource of spot evaluation.

Figure 5.10: Accuracy of PlaceEngine [19]

Every time the user selects a spot to visit, the Burari UI Module will send the id of the spot to the Activity manager. Then the User activity manager will memorize the spots in the order and will be grouped by date that the spot was selected. Each time a new spot was selected by a user, the User activity manager will send an update to the Burari spot manager in the form of selected spot id, a spot id list of today, and a list of all spot since the user started to use the service.

5.5 Burari Spot Manager

Burari spot manager consists of three modules: Spot sorter, Spot evaluator, and Spot pool. When a Spot Manager receives an update, the Spot pool will updates its list of visit-able spots depending on the current location of the user and remaining time slot.

Then the Spot evaluator and Spot sorter will evaluate each spot using the Time-aware Collaborative Sorting which will be described in the followings.

Time-aware collaborative sorting is a method that sorts a list of spots into the order that is assumed to be appropriate for the user's preference and context.

Figure 5.11 illustrates the mechanism and flow of the Time-aware collaborative sorting.

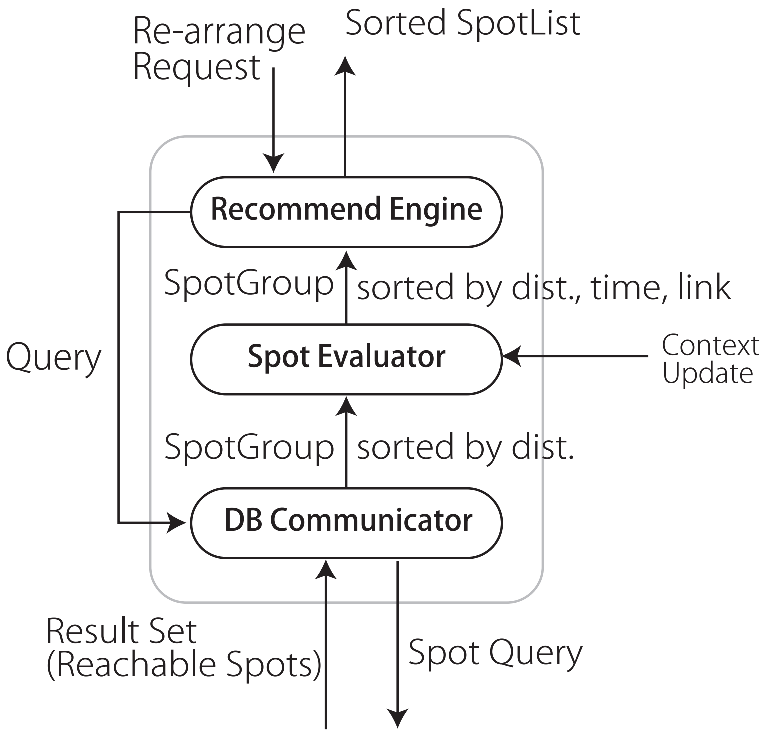

Figure 5.11: The flow of Burari Spot Manager

Time-aware collaborative sorting consists of three steps: pooling spots, evaluating spots, and sorting spots. Each step will be described at the followings.

5.5.1 Pooling Spots

First, the spots will be pooled into spot pool. When the spot pool receives an update message from one of the modules in Burari user manager, Spot pool will start updating the pool based on the current context of the user and then pass the spot group to the spot evaluator.

The policy of pooling the spot is straight forward. If the user could go back to the end location after visiting and spending a required time at a specific spot, the spot should be pooled.

The policy could be expressed in a pseudo query that would be a base concept of throwing query to the User spot database.

$SELECT * FROM\ spotdb\ WHERE\ ((spot\ loc - curr\ loc)/speed + reqtime) < remain\ time;$

5.5.2 Evaluating Spots

Next the Spot evaluator will evaluate the spots based on distance and the number of users who have visited to the spot from the previous spot.

Spot evaluation has two axes: Distance and Link Strength

- Distance
  Distance is a distance to the evaluated spot from the current location.
- Link Strength Link Strength is a number of users that have visited to the evaluated spot from the current or previous spot that the user has visited.
- (Similarity) Based on tag information attached to each spots, the system could evaluate the similarity of the spot between two spots. This function is not implemented yet.

Distance and Link Strength axis could be used in three styles depending on what the user choose at the grid browser. Three styles are Distance, Link Strength, and Mixture. Mixture will have an adjustable weight on each axis to adjust the evaluation function.

5.5.3 Sorting Spots

After evaluating the spots, the Spot sorter will sort the spots including redundant avoidance policy based on categories and tags to the evaluated spots. The redundancy avoidance is a policy to avoid recommending spots that would be redundant and must be avoided. For example, recommending a restaurant right after visiting a restaurant is a redundancy that must be avoided. On the other hand, recommending a similar clothing store right after visiting a clothing store might be helpful for a user. The avoidance should be implemented at the Spot sorter but the second example should be enabled at spot evaluation phase.

Therefore, the spot sorter will only care the avoidance case. The current implementation of Burari-navi only avoids redundant suggestion of Restaurants which is the most obvious case to avoid.

5.6 Databases

There are two databases in Burari-navi system. Burari spot database manages all spots information such as name, location, description, and meta-data which will be described in following of this section. User activity database manages selection history of all users including relationships of each spot in the selection history. Burari spot database is based on MAPPLE Guide Data [21] which is provided by Shobunsha which is a data of restaurant, amusement spots, tourist's spot and many other Point of Interests.

Burari spot database and User activity database are both implemented on the same database server running Postgresql with PostGIS [22] which enables fast and flexible query based on geographical information.

Figure 5.12 describes the schema of database and example values in Burari spot database and User activity database. Guidegis table is the table that holds data of spots including name, description, photos and others Keyword table is the table that holds the list of keywords. Spotkey table holds the mapping data of spots and keywords which means it tells which keywords are used for a specific spot.

5.6.1 Meta Information

In addition to information such as name, category, location, description, etc. the Burari spot database includes meta data to each spots so that the system could evaluate the spots from many aspects and more efficiently.

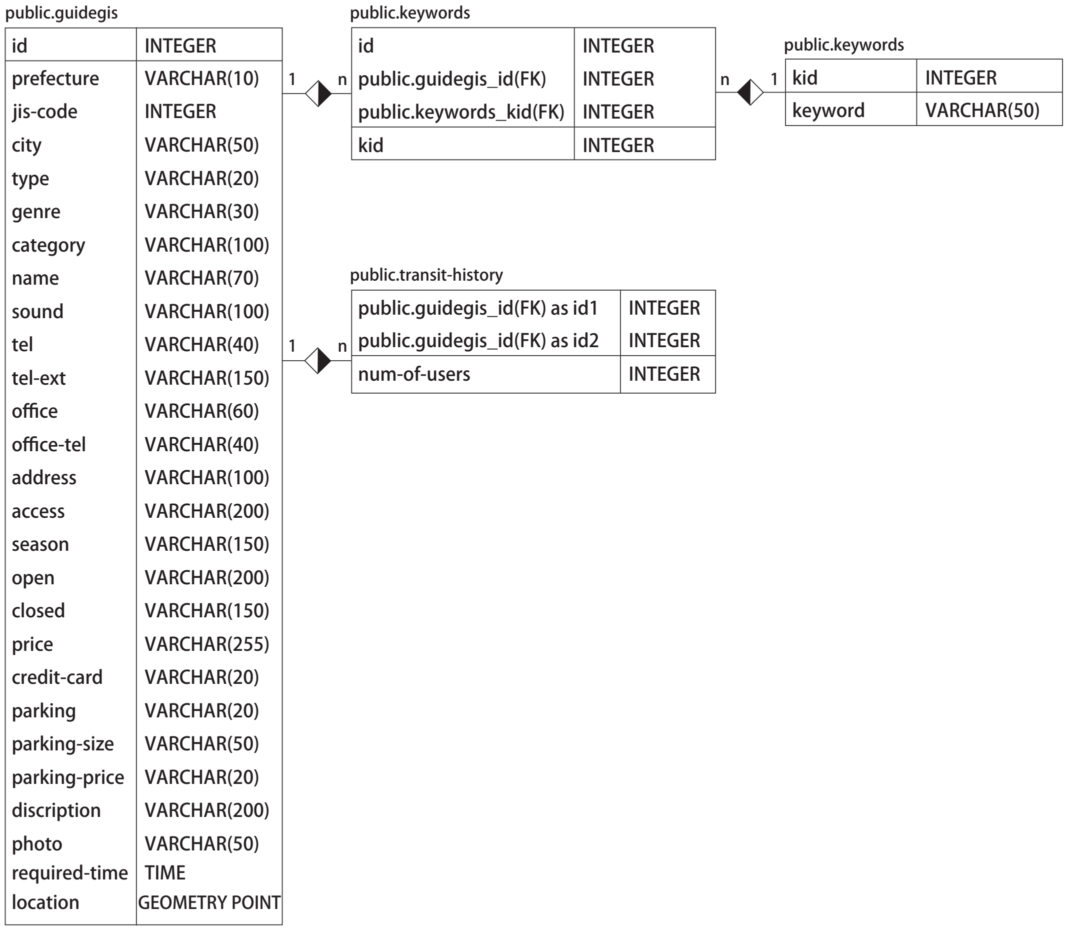

Figure 5.12: The scheme of Spots & Activity Database

- Tags

  Tags are a collection of keywords that represents the spot. For example, coffee shop could be categorized as coffee, restaurant, and/or rest. And tags could be more highly abstracted words such as “warm”,“calm”, “delicious”, and others. Currently, we do not use Tags for spot evaluation, but it could be used for algorithms such as Context-aware Collaborative Filtering. The tags could be created and updated by either the owner of the system or users.

- Required Time-slot

  Required time-slot is a minimum time to stay at a specific spot to enjoy. For example, a movie theater might be set three hours as a required time-slot. But if the movie theater has a coffee shop in it and it is valuable for users, the movie theater’s required time slot could be 15 minutes instead.

  Currently the minimum time-slot is set by the system owner based on categories, but it could be possible to give rights to the owner of the spots and users to update the required-time slot.

5.6.2 History Management

User activity database store and arranges users’ selection history into spot-to-spot link model which is the schema that the database holds inside.

The number of users visiting a mandatory spot to another spot are recorded for each combination and separated into four range of time: morning, lunch time, afternoon, and night. Current implementation does not separate history by time.

5.7 Map Service

Map Service will receive a request to show a map and routes based on location information provided from the user interface. There are two main function of the Map service [23].

One is to provide a map that shows the surrounding area of the user's current location. And another is to provide a route guide for given source location and destination location.

In current implementation of Burari-navi, we use a web browser based map service Google Maps [15] due to a limited time for a development. Map service could be easily replaced since the interface is designed in a simple manner: the system only needs to pass the longitude and latitude of the source location and destination.

5.7.1 Google Maps

Google Maps provides a function to display a map that contains both source and goal location displayed and connected by an arrow on the map as shown in Figure 5.13. There is no specific route guidance, but since there is a detailed map and both locations displayed on it, users are expected to reach the goal by their own just using Google Maps.

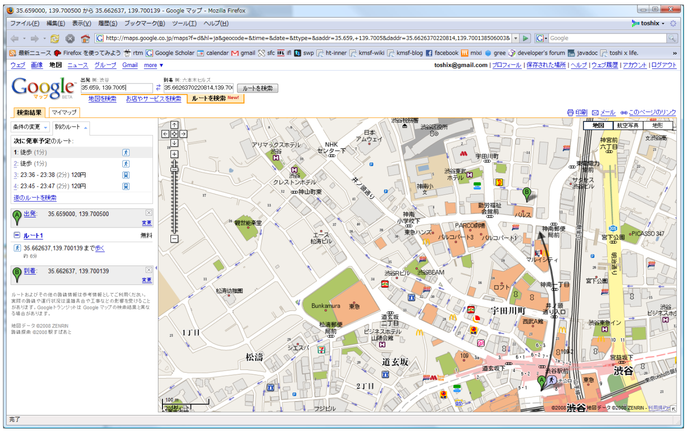

Figure 5.13: Screenshot of Google Maps

5.7.2 Map Server and Routing

We are looking forward to design and implement our own Map server. Currently, we have implemented a Map server which gives the map image of a mandatory area. Since we own a routing network data for pedestrian, we believe that the routing function could be better than Google Maps.

5.8 Summary

In this chapter, we have described Burari-navi from system's perspective including a description of the design and the current implementation of hardware and software components.

From the hardware perspective, Burari-navi consists of four parts: central mobile device, location sensor, spot and activity database, and map service. Currently, we use Vaio type-U as the central mobile device, PlaceEngine and GPS as the location sensor, Mac Mini as the database, and Google Maps as the map service.

Burari-navi consists of five software components: Burari UI, Burari User Manager, Burari Spot Manager, Databases, and Map Service. We have implemented the Burari UI, the Burari User Manager, and Databases fully functioning. The Burari Spot Manager and Map Service are partially implemented as a prototype.

We have done experiments using the current implementation of the Burari-navi system and we demonstrate the effectiveness of the system at the next chapter.

Chapter 6

Evaluation

In this chapter, we demonstrate the effectiveness of Burari-navi by evaluating the system from two perspectives: attainment level of the user's QoS and the system's usability.

6.1 Attainment of the QoS

In this section, we evaluate Burari-navi from the perspective of the QoS by running an experiment based on the scenario described in Chapter 4 and collecting users' impressions and reactions of using Burari-navi on the scenario.

6.1.1 Experiment

Collecting a feedback from the user via experiment is a necessary process of evaluating an interactive system. To collect the feedback from users and evaluate the system, we have run an experiment based on the scenario described in Chapter 4 for 13 participants. Participants were 18-23 years old and there were 10 male and 3 female.

First, we explained the purpose of the system and did a walk through-demo to teach what is Burari-navi for and how to use it.

Next, we explained the scenario which the user should experience to make the user imagine what kind of a situation is user located. The scenario that was used for the experiment is same as the scenario explained in Chapter 4. The user will be standing in front of Shibuya station around 1PM, the user does not have any verbal desire or purpose to do something at that moment. And the user has Burari-navi being launched in their hand.

Then, we let the user use Burari-navi for 5 minutes with-out guidance and collected the feedback using a simple survey which is attached at the end of this thesis. The survey had 31 question related to the QoS and the system's usability. Detailed information of questions would be explained in following part of this section.

The appearance of the experiment is shown in Figure 6.1

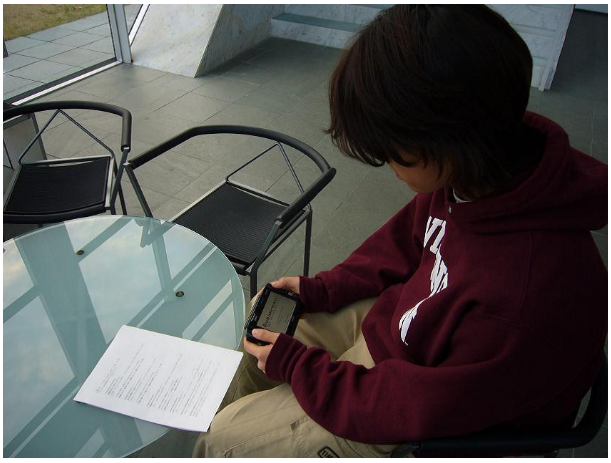

Figure 6.1: Scenario based Experiment

All processes of the experiment including the system was done and built in Japanese since all participants' native language were Japanese which is the efficient and loss-less language to communicate and collect feedback from participants. Questions and results are described in followings were translated into English by the author after the experiment.

6.1.2 User QoS Evaluation

First, we focus on how the system has fulfilled each components of the user's QoS guideline that we have defined in Chapter 2. And also we assure that the guidelines are accepted by the participants.

S1: Selection Completion

Being able to complete selection is the least required capability of the system. All participants were able to find and select an interesting spot for them and this shows that Burari-navi has cleared this requirement.

The feedback from participants shown in Table 6.1 shows that all users were able to select a choice.

Table 6.1: Results on Completion of Selection

Completion of Selection:	1	2	3	4	5	Average
Q1: Completed selecting and deciding a choice.	0	0	0	5	8	4.61

S2: Quality of Choices

Did the user satisfy with the choice or not is another important factor that supports the quality of choices.

The satisfaction level of choices has a wide range of results as shown in Table 6.2.

One reason of being disappointed was the gap between the impression of distance on the browsing interface and the actual distance shown on descriptions. This could be solved by working on a further survey on what kind of pictures were easy to understand or gave an inspiration to the user. Another reason was that some user felt that "the quality of spot cannot be evaluated without visiting". An on-site experiment should clarify this point and would be the next step for this research.

Table 6.2: Results on Quality of Choice

Quality of Choice:	1	2	3	4	5	Average
Q2: It was a satisfiable choice.	0	2	1	4	6	4.07

S3: Cost of Selection

As shown in Table 6.3, most of the users felt the cost of selection process was acceptable. But, some users had a hard time pointing the choice and the photo card expression did not match for some users. The problem with operation could be solved by learning and getting used to the interface as shown in usability evaluation of the browsing interface.

Table 6.3: Results on Cost of Selection

Cost of Selection:	1	2	3	4	5	Average
Q3: Selection process was tough.	7	0	2	3	1	2.30

S4: Worry of Return

Since one of the functions of Burari-navi is to manage the time of the user and support the user transiting to the next schedule on time, we evaluate how well has Burari-navi have done with supporting the user.

The result shown at Table 6.4 shows users are positive with letting the system manage the time and the projection of the remaining time is helpful.

Table 6.4: Results on Worry of Return

Worry of Return:	1	2	3	4	5	Average
Q4: I worried how much time do I have left.	2	2	2	6	1	3.15
Q5: Management of time by the system is helpful.	0	1	2	3	7	4.23
Q6: It is good to have remaining time to be shown on UI.	0	0	1	2	10	4.70

U1: Freedom of Activity

How free did the user feel using the system and was it comfortable or not is another aspect of the QoS.

The results shown in Table 6.5 indicates that users were not overwhelmed by a large number of results and felt that they had a control of what to look and what to choose. Also the policy of non dropping is supported by users on Q9.

Table 6.5: Results on Freedom of Activity

Freedom of Activity:	1	2	3	4	5	Average
Q7: I felt that the choices are limited/filtered.	8	3	0	2	2	1.69
Q8: Large number of choices were disappointing.	8	3	0	1	1	1.77
Q9: Large number of choices are good thing to have.	0	1	0	5	7	4.38
Q10: Some choices must be filtered (dropped).	7	2	0	3	1	2.15
Q11: Choice must be aligned/sorted in order obviously.	5	4	2	1	1	2.15

U2: Serendipity

We have defined an unexpected choice will be a factor that supports the QoS. The question is "Did the user encounter to unexpected, but interesting results?".

The result shown in Table 6.6 supports the needs of unexpected choice in exploratory-walk user. Users accepted mismatched choices and did not feel uncomfortable with them.

On the other hand, there were also participants that wants accurate suggestions. But since most of the users agreed that it is fun to have unexpected choices, an additional surface with accurate suggestion could be a solution.

Table 6.6: Results on Fun of Unexpected Choices

Fun of unexpected choices:	1	2	3	4	5	Average
Q12: I only want choices that matches my preference/activity pattern.	4	3	0	2	4	2.92
Q13: It is fun to have a result that does not match to my preference/pattern.	1	1	0	7	4	3.92

U3: Pleasure of Selection

Large group of choice should let the user feel that he/she has selected a bettter choice rather than selecting a choice from a small group.

The results shown in Table 6.7 proves the effectiveness of showing a large number of results to increase the QoS. We were expecting that some users might say that large group makes the user get confused, but fortunately, we did not get such result. Few participants pointed out that the brows-ablity of the Surface Browse UI is a factor that made the result positive. Also, there was a user who worried about what if the choices get larger and the memory on the device runs out, but this could be solved by implementing a robust image loading mechanism on user interface so that it will not load choices that are far from the area shown on display.

Table 6.7: Results on Pleasure of Selecting

Pleasure of Selecting:	1	2	3	4	5	Average
Q14: I feel satisfied by choosing a choice from a large group.	0	0	0	4	9	4.69
Q15: It is good to have many choices.	0	0	0	6	7	4.53

6.1.3 Requirements Achievement

We have also collected feedback on how did the prototype achieve the requirements for exploratory-walk navigation systems.

Inspiring UI

The first requirement was an inspiring user interface that supports users to figure out what are their desires hidden inside.

Results shown in Table 6.8 supports that users felt that current implementation of the user interface is effective with giving inspiration to the user. Users pointed out the number of choices, the grid layout, and the Photocard impression as the main factor of giving inspiration to the user rather than sorting.

Photocard expression was supported by users but there were some cases that the user felt a gap between the impression and the description of the card. This problem must be solved by controlling the quality of data that are store in the database. The quality control will be an important aspect of future works.

Table 6.8: Results on Inspiring UI

Inspiring UI:	1	2	3	4	5	Average
Q16: This user interface supports finding interesting choice with out purpose.	0	1	0	2	10	4.69
Q17: This user interface gives an inspiration to a user.	0	0	0	5	8	4.61
Q18: Photocard Expression is effective than text based information.	2	0	0	4	7	4.07
Q19: There were a gap between the impression and description on photocard.	0	0	1	5	7	4.46
Q20: Three types of surface was effective and useful.	0	0	0	4	9	4.69

Robust and Flexible Flow Management

The flow design of the Burari-navi system prototype was mostly supported by the user as shown in Table 6.9. Further evaluation on flow management and usability will be done at Section 6.2.

Table 6.9: Results on Robust and Flexible Flow Management

Robust and Flexible Flow Management:	1	2	3	4	5	Average
Q21: The flow of the system were easy to understand and use.	1	1	0	6	5	4.00

Time and Location Management

Time and location management were also mentioned as a requirement. Results shown in Table 6.11 indicates an obvious needs of both time and location management, but not many obvious requests of further functions for both factors. Also the result on Q26 supports the design of system flow letting the user concentrating on selecting a choice rather than worrying their position.

Table 6.10: Results on Time and Location Management

Time and Location Management:	1	2	3	4	5	Average
Q22: It is good to have remaining time displayed.	0	0	0	5	8	4.61
Q23: Not noticing the remaining will make me worry.	0	1	0	4	8	4.46
Q24: If the map is available, time management is not necessary.	0	4	0	7	2	3.53
Q25: I can visit the choice with current implementation of guidance.	3	0	2	6	6	3.61
Q26: The guidance function of EZ Navi Walk and NAVITIME could be used.	1	1	0	3	8	4.23
Q27: I want a map shown all the time on display even when selecting.	5	3	4	1	0	2.07

6.2 Usability Evaluation

In this section, we evaluate the system from the perspective of usability based on an experiment and demonstrate the degree of achievement level on interaction design.

6.2.1 Experiment

We ran a usability test to evaluate the achievement level on user interface interaction design and figure out the hint of further achievements. The evaluation will be done by following the 5 indicators [24, 25]: learnability, efficiency, memorability, error handling, and user satisfaction.

We let the user to find and select a specified spot which is located in the group of choice on the user interface and measured the time from the moment that the user started browsing to the moment that the user selects the specified spot. We always specified a spot which is placed 5 spot away from the participant's current location on the surface so that the distance and cost is same through all tries and all participants.

We had 6 participants and who are 18-25 years old and have only used Burari-navi once for previous evaluation experiment.

Before starting the experiment we requested the participants to explain how to operate the UI to check if the users remember the user about how to use the interface. Next, we let participants to repeat the selection task 5 times and measured the time for each try as shown in Figure 6.2.

Figure 6.2: Usability Experiment

6.2.2 Results

Figure 6.3 shows the result of the experiment.

All participants were able to select the specified spot within 3-7 seconds except 2 tries which took about 10-12 seconds. Participants said that "there were no stress with selecting the spot and it was easy to use from the first time" and the result supports the opinion since all users did not take any extra time at the beggining of the experiment.

In the following of this section, we discuss Burari-navi's usability from five perspectives: learnability, efficiency, memorability, error handling, and user satisfaction.

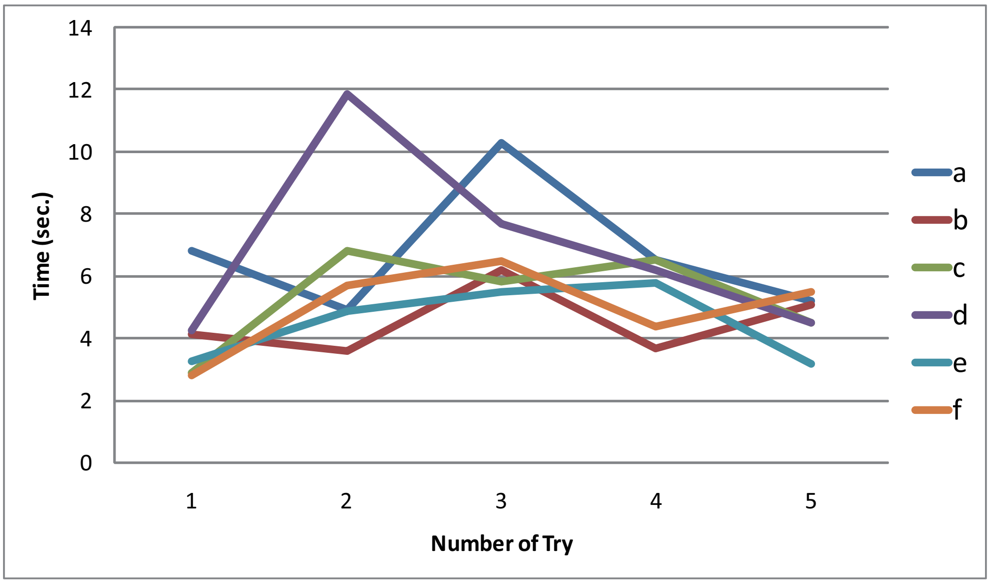

Figure 6.3: Elapsed Time to Select a Mandatory Spot

Learnability

For each participant, there were no obvious difference of the elapsed time at the beggining of the experiment. This indicates that the Burari-navi's UI has significantly low cost for learning. And all participants agreed that the selection process was easy and speedy enough. The Burari-navi's UI has enabled low cost learning and ease of use.

Efficiency

The result shows that the all participant could select the specified spot in five seconds for average. As the result from the QoS evaluation indicates that the selection task was not tough, five seconds seems to be an acceptable time. This demonstrates the achievement of efficiency in the interaction design of Burari-navi.

Memorability

All participants remembered the operation method to browse through the surface. Since the operation interface is only a touch screen and four extra buttons, it was easy to remember for participants.

Error Handling

According to participants, there were few errors that happen to occur but all of them were easy to recovery by moving the focus in other direction.

Most critical error of the operation interaction was when the participant's finger overlapped to the center of the screen since it interferes to check the position of the focus and spots while selecting. Another error was when the focus of the interface dropped in between two spots, but this was also easily recovered by dragging the screen.

Some errors were found but the errors could be easily handled by detailed orientation, learning, and small adjustment on UI parameters.

User Satisfaction

As the results shown in Section 6.1 and in Table 6.11, most of the users were satisfied with the appearance and interactions made via the UI and mentioned that they would like to use it again in a real situation.

Table 6.11: Results on User Satisfaction

User Satisfaction:	1	2	3	4	5	Average
Q28: I would like to use this system in daily life.	0	0	0	2	8	4.80
Q29: This system is useful for exploratory-walk.	0	0	0	1	9	4.90
Q30: Exploratory-walk occurres in daily life.	0	0	1	1	8	4.70
Q31: Burari-navi is effective than NAVITIME and EZ Navi Walk.	0	0	2	5	3	4.10

Also, there were a large number of comments made on the interaction by participants such as "I want it implemented on cellular phones" and "I would like to see more collaboration with existing services such as Gourmet Navi". This shows the expectation of improvement and excitement of using the Burari-navi system. And for further research and improvement, an on-site experiment is desired.

6.3 Summary

In this section, we demonstrated the effectiveness of Burari-navi by evaluating the Burari-navi from the perspective of the QoS achievement level and system usability by running two experiments.

At the first experiment, we let participants use the prototype of Burari-navi. Most participants were satisfied with the QoS attainment level of the system, but we also gained feedbacks which enlightened us for further improvement such as "The quality of spot cannot be evaluated without visiting" and "There were a gap between the impression of the thumbnail and description of the spot".

The second experiment was done to demonstrate the usability of the Burari-navi system. The results illustrated the effectiveness of using a consistent operation interface and also the high learnability of the Surface Browsing User Interface. Also, results and comments made by participants demonstrated the effectiveness and a high level user satisfaction of the Burari-navi's basic functionalities.

Chapter 7

Related Works

In this chapter we introduce and analyze related works from three perspectives:navigation systems, browsing interfaces, and spot filtering methods.

7.1 Navigation Systems for Exploratory-walk

7.1.1 Everyone's Map

Everyone's Map [5] is a map-information-software released from ZENRIN for PlayStationPortable which is sold from Sony Computer Entertainment Inc.. It provides area information, route guidance, evacuation route in case of disaster, some trip plans for users. It locates the user using both PlaceEngine and GPS and also enables users to share information via a website called PetaMap. The system has a broad coverage of users and exploratory-walk users might be able to use as well, but the brows-ability of the information is less than Burari-navi, since it only has map based browsing user interface.

7.1.2 TSUGI DOKO

TSUGI DOKO [26] is a service that is currently on an experiment phase which is run by NTT docomo. TSUGI DOKO is a push-type service that recommends sites that might be interesting for users depending on users' location and profile. Recommendation is done by using current location,

Figure 7.1: Everyone's Map by ZENRIN

profile, and other users' activity history.

Since TSUGI DOKO is a push-type service, the user does not have a free to browse through other candidature sites and also the user interface lacks the brows-ability. The way TSUGI DOKO manages the user profile are accurate because it requires users to input their profile at the first time use but on the other hand it becomes a cost for a user to start the service. Also, TSUGI DOKO does not have Time Management function which is required by exploratory-walk users. A use image of TSUGI DOKO service is shown at Figure 7.2.

7.1.3 COMPASS

COMPASS [27] is a context-aware mobile tourist application that provides information that is predicted to be needed for a user depending on the context of the user. COMPASS is built on the WASP platform [28] which supports managing context and Point-of-Interest (POI) database. Users were required to input their preference and the system managed their activity history to avoid redundant recommendation such as recommending another restaurant after visiting a restaurant. COMPASS has two types of recommendation. One is based on similarity of the user-profile and POI

Figure 7.2: Use Image of TSUGI DOKO Service

characteristics, and another is redundancy avoidance using activity history.

An evaluation of recommendation engines done in the paper shows that less than fifty percent of the people felt either function useful. And according to the paper, some participants said that "I can think and decide for myself". This indicates that there are users that does not prefer narrowed recommendation and rather would like to get more choice. Even though the COMPASS had implemented a mobile tourist application with intelligent recommendation, it was not completely supported by users.

And also, COMPASS does not have time management function and brows-ability since it was built on an old device and focused on smart recommendation.

7.2 Browsing Interface

7.2.1 InfoGlobe

InfoGlobe [29] is a push-based interface purposed to support exploratory users of an online bookstore such as Amazon. The purpose of the system is to let the user browse a large number of books mapped on a sphere floating in a virtual space. The user interface will move the choices on

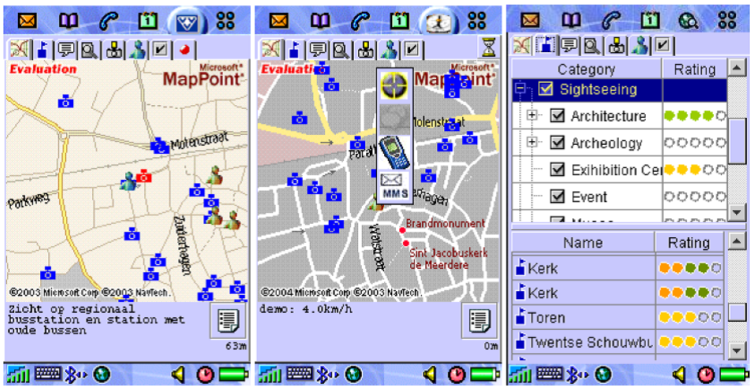

Figure 7.3: Screenshot of COMPASS System

the display without users operation and this lowers the cost of selection, but on the other hand this does not create a pleasure of selection and user might feel controlled by the system. Therefore, the main idea is similar but the approach has a different direction and Burari-navi has fulfilled the QoS factors. The screenshot of InfoGlobe is shown in Figure 7.4.

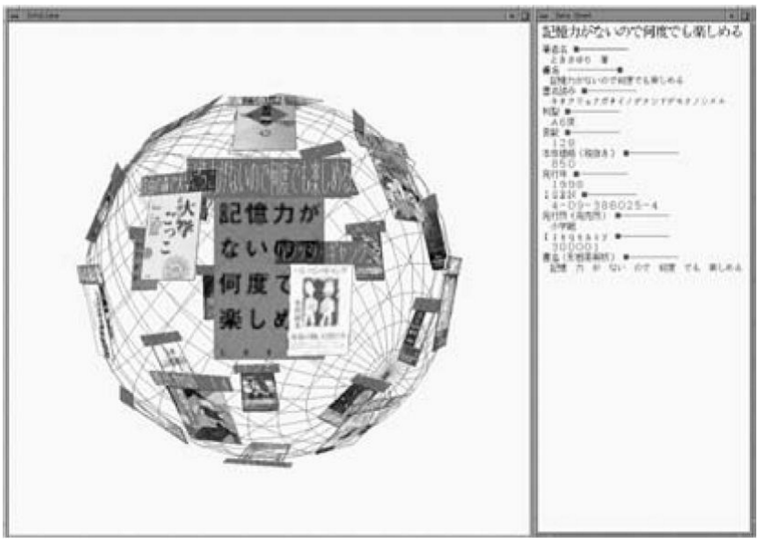

Figure 7.4: Screenshot of InfoGlobe

7.2.2 Gards

Gards [30] purposed by Otsubo is a recommendation system that adapts to changing human preference based on a user's input. Gards shows pictures of candidates in a three-by-three grid and let the user evaluate positive or negative on candidates and then show the next candidate sets corresponding to the preference input.

Figure 7.5 show how Gards acts to the user input. When a user select one picture as positive candidate, the system will rearrange the candidates so that it will match to user's preference at that moment.

In this case, the system can adapt immediately to the user's changing preference unlike other recommendation engine such as item-item filtering and collaborative filtering. The approach that Gards made are very similar to our approach, but Gards only focuses on restaurant's menu and also does not concern about location and time management. From the user interface point of view, Gards only provides the scale of three by three at once and this might be not enough for well trained users and must have some scaling mechanism.

7.2.3 INFOTUBE

INFOTUBE [31] implemented by Wakita et al. is a system that enables users to browse a shopping arcade with in a virtual three dimensional space. Shops are mapped to a surface of a tube according to physical location and logical connections or similarity as shown in Figure 7.6.

INFOTUBE enables user to browse through a large number of shops intuitively, but since the concept of INFOTUBE is visualization, the system is not yet completed as a service. The approach of the visualization is similar to our approach especially to the selection browser, but is a bit too complicated for both users and mobile device Therefore, it is an interesting approach from visualization point of view, but too complicated for exploratory-walk users.

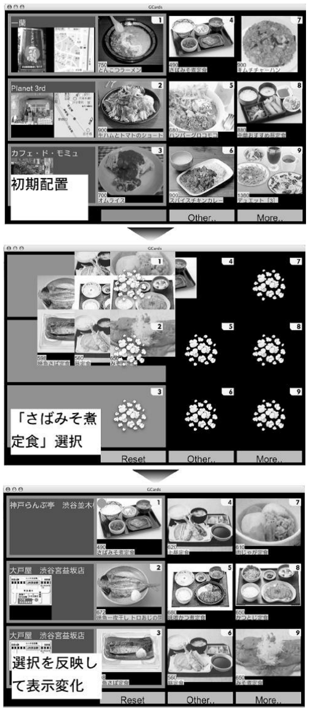

Figure 7.5: Screenshot of Gards System

Figure 7.6: Screenshot of INFOTUBE

7.3 Spot Filtering Method

7.3.1 Context-Aware SVM

Context-aware Support Vector Machine (C-SVM) and C-SVM with Collaborative Filtering (C-SVM-CF) [32] is a method that integrated a user's context as another axis to basic SVM engine for recommendation purposed by Oku et al.

Since C-SVM and C-SVM-CF, both recommendation engines require learning process to be functional. The problem is that, learning process is painful for the user and will take time to be functional. Also, in case of exploratory walk, user's preference and context is not clear and change easily. Therefore, it is not appropriate to define a rigid axis of context in SVM.

As an engine for subjective-walk model, it might be very useful but not for exploratory-walk.

7.3.2 Context-Aware CF

Another similar method called Context-Aware Collaborative Filtering System is proposed by Chen [33] which is an expansion of item-to-item collaborative filtering [34] and work is still in progress. This method has defined context as an activity history and purposed that the user that have a similar activity history must have the same next step in high possibility. But again, this method requires a number of user's history and since the exploratory user's goal and motivation changes frequently and the factor is almost unpredictable, it is difficult to prove that this method will work well with exploratory walk.

7.4 Summary

In this chapter, we introduced related works from three perspectives: navigation systems for exploratory-walk, browsing interfaces, and spot filtering methods. And pointed out both advancements and weaknesses of each representative work.

Each system has strengths in a specific view point such as portability, but has a critical weak points such as lacking the aspect of location or having a bad balance as an exploratory-walk navigation system because the exploratory-walk user model is not expected in each system.

Chapter 8

Conclusion and Future Works

In this chapter, we conclude this thesis by summarizing this thesis including overall conclusion and discussing the future directions of Burari-navi.

8.1 Conclusion

This thesis purposed the Burari-navi system which is a QoS-driven exploratory-walk navigation system. The system enables users to browse, find, visit, and enjoy without any explicit verbal desire and preference inputs except time and location restrictions from the user. First of all, this thesis purposed and created a new style of navigation system which is called exploratory-walk navigation system. An exploratory-walk user model is a new category of users that was not supported by existing navigation systems. Then we analyzed and purposed axes of the quality of satisfaction of exploratory-walk users. And defined the requirements for the exploratory-walk navigation systems including a clarification of problems with existing navigation systems.

We have designed Burari-navi which meets the requirements and increases the QoS of users by providing a large number of choices via a high brows-able user interface. The user interface covers the dilution of the quality by increasing the brows-ability, using an inspiring rich picture card expression, and applying a sorting to the choices to increase the efficiency. The selection of the choices does not have any dropping process to avoid oversight. Sorting is based on time, transition probability of all users, and redundancy check.

We described our current implementation Burari-navi which focuses on user interface and interaction. And we demonstrated the fulfillment of the requirements and level of the QoS by evaluating the Burari-navi based user feedback on a simulation experiment and a usability evaluation as well. We assured that participants' QoS were satisfied and requirements are fulfilled overall. Most participants said that they would like to use the system again in a real situation and this showed the level of user satisfaction from the perspective of system usability. Comments made by participants such as "there were a gap between thumbnail and description" and "the quality of spot cannot be evaluated without visiting" indicated needs of further improvements and onsite experiment.

And finally, we will discuss few key aspects for further research on Burari-navi and other exploratory navigation systems.

8.2 Future Work

This section suggests directions for future research. With the Burari-navi, three issues remain unresolved: managing the quality of spot database, optimizing the guidance phase, and implementing Burari-navi on cellular phones.

8.2.1 Managing the Quality of the Spot Database

Quality of suggestions made by Burari-navi strongly depends on the quantity and the quality of spot information stored on the database. Since the current implementation of Burari-navi is using a spot database based on data from tourism and restaurants guidebooks, the database has a slight category bias. User-driven data management and advertisement integration are two possible solutions to provide a service equally to various kinds of users and improve the quality of suggestion.

User-driven Data Management

The shortest way to make a database for a user is to let the user make the database. Users know their own needs and the value of the information. Especially when there are users with a variety of characteristic, it is much efficient to collect information from users rather than managing the information by a small group of administrators. Information collected by users could change the database in either a good or a bad direction. To lead the database to evolve, there should be a mechanism that supports users to input information efficiently and the system should have a robust integration method. The user-driven data management has a chance of increasing the number of database entries and deleting the bias made by the system administrator, but also has the risk of minority oversight and overflow of information. Therefore, we look forward for an approach that has a good balance of user-driven factor and owner administration for improving the quality and quantity of information.

Advertisement Integration

Since most of the information provided in urban areas and tourism sites are related to industrial market, getting industries on our side is an important process for improvement and deployment of Burari-navi. If we could design a business model on allowing ads as a single spot or a part of a spot, its income should support the system deployment and also updated information such as pruoduct information could be useful for improving the quality of the data. But on the other hand, business could cause a bias on spots and also this is only applicable for urban areas and tourism sites. This could be another aspect of evolving Burari-navi and the database.

8.2.2 Optimizing the Guidance Phase

Our current implementation is using Google maps, which is an open map service on web which has high accessibility but poor functions. Since the current version of Google maps does not have routing function for pedestrians, pedestrian optimized routing function is needed most to provide the minimum required service. There are few existing services such as EZ Navi Walk, which has routing and guidance function for pedestrian and those are current candidates to collaborate with. But to provide a best navigation phase for exploratory-walk, further analyze of transport phase of the exploratory-walk user and unique implementation is needed. For example, the exploratory-walk user are always looking forward for better choice of next and therefore might be waiting some suggestion even during walking. In this case, a solution might be to continue suggesting even on the map using the area browser with map. Consequently, a further research on guidance phase could improve the Burari-navi.

8.2.3 Implementing Burari-navi on Cellular Phones

Another future work is implementing the Burari-navi on cell phones. We have designed the user interface expecting touch-screen to become a standard user interface on cellular phones as some smart phones such as iPhone already have touch-screen on it. Gaps with the current ability of cellular phones with the current implementation are the limited CPU power, low screen resolution, and limited battery-life. But all of these technologies could be expected to be solved in next few years since all of them are increasing rapidly in the current market. The implementation on an off-the-shelf product accelerates the deployment and we are aiming at it as our next step.

Acknowledgments

First of all, I would like to thank my advisor, Professor Hideyuki Tokuda. Professor Tokuda always took time out of his extremely busy schedule to support, guide, and encouraged me at both home and campus.

I would also like to thank Professor Kazunori Takashio, Dr. Jin Nakazawa, Dr. Masayuki Iwai, Junichi Yura, and Masaki Ito for their daily supports and valuable advices.

And I am especially grateful to Mizuki Kawazoe, Katsuya Hashizume, Tomotaka Ito, Kyohei Kawada, Naoya Namatame, Yoshi Tokuda, Jumpei Yamamoto, Takuya Takimoto and all other members of "Keio Media Space Family" and "i208" for providing great support and stunning insights while I was writing this thesis.

Huge thanks to all the members of Hide Tokuda & Kaz Takashio Laboratory for making my life at the laboratory filled with fun and excitement.

Last and most importantly, I would like to thank my family for encouraging and supporting me all the time.

February 7, 2008

Hidetoshi Tokuda

Bibliography

[1] L. Meng, A. Zipf, and T. Reichenbacher. *Map-based Mobile Services: Theories, Methods and Implementations*. Springer, 2005.

[2] E. Kaasinen. User needs for location-aware mobile services. *Personal and Ubiquitous Computing*, 7(1):70–79, 2003.

[3] Inc. au. EZ Navi Walk. `http://www.au.kddi.com/ezweb/service/ez_naviwalk/`.

[4] NAVITIME Japan. NAVITIME. `http://www.rurubu.com/`.

[5] LTD. ZENRIN CO. Map Software: Everyone's Map. `http://www.rurubu.com/`.

[6] M. Hazas, J. Scott, and J. Krumm. Location-aware computing comes of age. *Computer*, 37(2):95–97, 2004.

[7] Hidetoshi Tokuda, Masaki Ito, Kazunori Takashio, and Hideyuki Tokuda. Burari-navi: an encounter oriented pedestrian nvigation system. In *Multimedia, Distributed, Cooperative and Mobile Symposium (DICOMO2006)*, pages 485–488. IPSJ, 2006.

[8] Pioneer Corp. Cyber Navi: Car Navigation System. `http://pioneer.jp/carrozzeria/cybernavi/`.

[9] Ltd. Matsushita Industrial Electronic Co. Strada: Car Navigation System. `http://panasonic.jp/car/`.

[10] CyberMap Japan Corp. Map Information Retrieval Site: Mapion. `http://www.mapion.co.jp/`.

[11] Catalog Inc. Travel Planning Site: Tripl. `http://tripl.to/`.

[12] LTD. RECRUIT CO. Super Map. `http://sugoi.doko.jp/`.

[13] GOURMET NABIGATOR INC. Gourmet Navigator. `http://www.gnavi.co.jp/en/`.

[14] LTD. RECRUIT CO. Gourmet Site: HotPepper.jp. `http://www.hotpepper.jp/index.html`.

[15] Google Inc. Google Maps API -Google Code-. `http://code.google.com/apis/maps/`.

[16] J. Preece, Y. Rogers, and H. Sharp. *Interaction design*. Apogeo Editore, 2004.

[17] D.A. Norman. *Emotional Design: Why We Love (Or Hate) Everyday Things*. Basic Books, 2004.

[18] A.M. MacEachren. *How Maps Work: Representation, Visualization, and Design*. Guilford Pubn, 1995.

[19] J. Rekimoto, A. Shionozaki, T. Sueyoshi, and T. Miyaki. PlaceEngine: A wifi location platform based on realworld-folksonomy. *Internet Conference 2006*, pages 95–104, 2006.

[20] Inc. Apple. iPhone: Features. `http://www.apple.com/iphone/features/index.html`.

[21] Inc. Shobunsha Publications. MAPPLE Guide Data by Shobunsha. `http://www.mapple.co.jp/corporate/product/09.html`.

[22] P. Ramsey. PostGIS Manual.

[23] T. Mitchell. *Web Mapping Illustrated*. O'Reilly, 2005.

[24] J. Nielsen. *Usability Engineering.* Academic Press, 1993.

[25] J. Nielsen. *Usability inspection methods.* ACM Press New York, NY, USA, 1995.

[26] Inc. NTT docomo. "Press Release: Target Marketing Using Location Information (Japanese)". `http://www.nttdocomo.co.jp/info/news_release/page/071120_00.html`.

[27] M. van Setten, S. Pokraev, and J. Koolwaaij. Context-Aware Recommendations in the Mobile Tourist Application COMPASS. *Adaptive Hypermedia And Adaptive Web-Based Systems: Third International Conference, AH 2004, Eindhoven, The Netherlands, August 23-26, 2004: Proceedings*, 2004.

[28] P.D. Costa. Towards a Services Platform for Context-Aware Applications. *Telematics, UniversityofTwente, Enschede. Netherlands. August*, 2003.

[29] Minakuchi Mitsuru and Urano Naoki. Information Browsing System by Push-based Interface. *Workshop on Interactive Systems and Software 1998*, 98:159–164, 1998.

[30] Otsubo Goro. Gards - Information recommendation system which assumes ever changing human preference. *Proceedings of the13th Workshop on Interactive Systems and Software*, pages 31–36, 2005.

[31] Akira Wakita and Fumio Matsumoto. Information visualization with web3d: spatial visualization of human activity area and its condition. *SIGGRAPH Comput. Graph.*, 37(3):29–33, 2003.

[32] Oku Kenta, , Nakajima Shinsuke, Miyazaki Jun, and Uemura Shunsuke. Context-aware SVM for Context-Dependent Information Recommendation. *DEWS 2006*, 1A-oi2, 2006.

[33] A. Chen. Context-aware collaborative filtering system: Predicting the user's preference in the ubiquitous computing environment. *Proc of CHI'05*, 2005.

[34] G. Linden, B. Smith, and J. York. Amazon. com recommendations: item-to-item collaborative filtering. *Internet Computing, IEEE*, 7(1):76–80, 2003.