TITLE

A METHOD AND RELATED GRAPHIC USER INTERFACE (GUI) DESIGN FOR PRESENTING CARPARK INFORMATION

FIELD OF THE INVENTION

The present invention relates to a new method and related GUI design for presenting car parking information simultaneously, such as carpark name, location, tariff and lots available on a single display screen to facilitate the choice of a suitable carpark near to the travel destination.

BACKGROUND OF THE INNOVATION

One of the main functions of a carparking related mobile application is to provide carpark information to users for choosing a carpark that better fit their parking needs.

In order to chose a carpark for parking the vehicle used, users normally consider the following aspects of carpark information:

1) The location of the car park - it shall be a carpark that is located as near as it is acceptable to a target location (the location where the user intends to go);
2) The carpark must have available parking lots/spaces for the vehicle used;
3) The parking charges shall be affordable (normally, the cheapest carpark is better for most users).

The carpark chosen finally by a user is dependant on the information the user can get about the car parks near to the user's travel destination and also on the balancing importance of the aspects 1) and 3) above to the user.

The location of a car park can be presented by carpark/building name, address and geographic coordinates (latitude, longitude).

The data of available parking lots/spaces are related to vehicle types. A carpark may have different numbers of parking lots/spaces available for different vehicle types, such as

car/van, motorcycle, lorry/heavy vehicle, containers, handicapped, electric car/van, and etc.

For a carpark and a certain vehicle type, eg. car, the parking charge is dependant on some or all of the factors below:

- the parking duration
- the type of day (Weekday, Saturday, Sunday, Public holiday)
- the time zone(s) in a day [a day could have a few time zones, e.g. 7am to 6pm, 6pm to 11pm, and 11pm to 7am (next day)]. Normally the charge rates for different time zones are different
- the charge rates may be described as
  - per period charge (such as a rate for per minute charge, per 30 minutes charge, or per hour charge),
  - per entry charge,
  - per session charge with multiple entries,
  - charge(s) with limitation could be applied for a time zone and for a day.
- an entry grace period is normally applied for most carparks, and an exit grace period could be applied if the payment is not done at exit point.

How to present some necessary data from so much complicated car parking information in order to help users to easily find suitable car parks that meet different users' different needs?

Some mobile apps use a list of the name and address of the carparks near to the travel destination, then user has to click on the name of a carpark to see the related information of lots available and then view and understand the parking charge schemes. In this way, the user can only see the information for one carpark at a time - it is not easy to do a comparison among different carparks in order to choose a carpark that better fit the user's needs.

Another way of presentation of carpark information used by some mobile apps is to display carpark icons together with the car lots available - the user has to click on a carpark icon to see the carpark's other information, such as carpark name, and parking

charge rates. This is similar to the way of using a list of carparks. However, the user can only see the information of one carpark at a time, so that it is difficult to remember so much information for comparing.

Is it possible to present a few key parking data of all the carparks near to a target location together on a single display screen, so that a user can easily compare the data for finding the carpark that better fit his/her parking needs?

To manually compare parking charges among different carparks can be a difficult task. For example, for a single day, the parking fee charge scheme of two carparks nearby could be as shown below:

Carpark A:	08:00 to 18:00, $11/hour	Carpark B: 09:00 to 20:00, $18/hour,
	18:00 to 22:00, $16/hour	20:00 to 24:00, $10/hour,
	22:00 to 08:00, $4/entry	00:00 to 09:00, Free

How could the parking charge be presented in a single display screen? Which carpark is the cheaper carpark? The answer is not obvious. It depends on what time a user starts parking, and also on whether the parking period crosses any time zone.

Is it feasible to indicate what carpark's parking charge is the cheapest among the carpark displayed on the screen? The present art does not provide the solutions to these issues.

OBJECT OF THE INVENTION

The objective of this innovation is to provide a method and related GUI design to define and present together the necessary information of all the carparks near to a target location on a single screen in a manner of easy reading, understanding and comparing, so that it allows users to be able to quickly make a decision for choosing a carpark that better fit their parking needs.

SUMMARY OF THE INVENTION

In accordance with one aspect of the present invention, the necessary information of a carpark for a certain vehicle type is defined as the key parking information that include the following 5 key parking data:

1) the name/short name of the carpark/building;
2) the location of the carpark on the geographic map;
3) the current availability of parking lots/spaces of the carpark;
4) the current parking fee charge rate;
5) the end time of the current charge rate.

In accordance with another aspect of the present invention, the key parking information of a car park is displayed by a carpark icon together with texts surrounding the icon on a geographic map.

Preferably, the carpark icon is of a pin icon, however it could also be of other shapes. The location of a carpark icon on the map of a mobile phone screen is determined by the geographic coordinates (latitude, longitude) of the carpark.

In accordance with another aspect of the present invention, a carpark near to a travel destination (or target location) is defined by the inequality below

$$d \le d_i = i * d_0,\ i = 1, 2, \ldots$$

where

d is the the distance between a carpark and the target location. d can be calculated based on Haversine_formula by using the coordinates of the target location and of the carpark location.

d_0 is the preprogrammed initial radius value of the search area.

d_i is a variable radius of display area surrounding the target location.

In accordance with another aspect of the present invention, the carparks near to a target location can be found following the procedure below:

1) obtain the coordinates of the target location ($Ø_t$, λ_t)

2) set a limitation of search area $\Delta(Ø) \leq Ø_L$, $\Delta(\lambda) \leq \lambda_L$, the carparks within the limited area are with the coordinates $(Ø_j, \lambda_j)$, j=1, 2, ….,

where $\Delta(Ø) = (Ø_j - Ø_t) \leq Ø_L$, $\Delta(\lambda) = (\lambda_j - \lambda_t) \leq \lambda_L$

3) for i =1

4) take $d_i = i * d_0$,

5) find the carparks that meet the condition $d \leq d_i$

6) if the number of carparks found is less than N (N is a pre-programmed integer), then i = i +1, go to 4), else to display the key parking data of all the carparks near to the target location on a single screen.

This procedure ensures that at least N car parks near to the target location can be found, with reduced computation.

Preferably, the parking lots availability of a carpark corresponding to a selected vehicle type is defined by a range of available percentage (the number of lots available over the total number of the lots in the carpark). Preferably, at least (but not limited to) three availabilities can be defined as:

- low availability, if the available percentage is lower than or equal to a%,
- high availability, if the available percentage is higher than b%,
- middle availability, if the available percentage is between the a% and b%.

The value of a% and b% can be determined by the management of the car park, and be set in the database by a system support administrator.

Preferably the lots available levels are presented by using different colors to fill the upper portion (the round shape) of the carpark location icon. For example, Red color for Low availability, Orange color for middle availability, and Green color for High availability.

Preferably, a different color (e.g. Blue color) is used to indicate that there are some lots available for the selected vehicle type in the carpark but there is no percentage data available.

Preferably, another different color (e.g. Black color) is used to indicate the fact that the lots of available data of a car park are not available.

Preferably, the current parking fee charge rate and the end time of the charge rate for the selected vehicle type in a car park are displayed in texts at the position between the car park name/short name and the car park location icon.

In accordance with another aspect of the present invention, the cheapest carpark is defined as the carpark with the minimum charge for a certain parking time period (e.g. 1 hour) starting from the current time, among the carparks near to the target location, in the condition that at any time within the 1 hour parking period starting from the current time, the charge rates of all carparks must be the same type (e.g. either all are per hour charge or all are per entry charge). If at any time, within the 1 hour period from the current time, there are different charge rate types (e.g. one carpark is with per entry charge, other carparks are all with per hour charges), then the cheapest carpark is not applicable.

Preferably, a cheapest carpark sign is presented above the carpark name/short name of the car park(s) to indicate that the carpark(s) is (are) the cheapest carpark(s) among the carparks displayed on the GUI screen.

BRIEF DESCRIPTION OF THE GUI FOR AN EMBODIMENT

The present invention will now be described, by way of example only, for an embodiment that is a mobile app system. One of the features of the mobile app system is to provide carpark information to help users to easily find a carpark that better fit their parking needs.

The mobile app system consists of a cloud based backend database and application servers, and a front-end program with GUIs running on users' mobile phones.

Parking information of carparks is stored in the backend database. The parking information stored for a carpark include, at least but not limited to, the following:

(A) car park code/name/short-name, address, geo-coordinates and etc.

(B) the total number of parking lots and the percentages for defining parking lot availabilities;

(C) the parking tariff including all parking fee charge rates in every time zone for different day types and vehicle types.

Once a user inputs/searches/selects a target location, the system will firstly find all the carparks near to the target location by using the distance condition of $d \leq d_i$; secondly, it retrieves the key parking data for each of the carparks found; thirdly, it finds the cheapest carpark(s) if the condition of the same charge type is applicable; and then presents the key parking data for all the carparks, as well as the cheapest carpark sign (if applicable) together, on the single display screen.

By comparing the key parking data of the carparks on the single screen, the user can chose a carpark that is better fitting to his/her parking needs.

Reference to the accompanying figures of the embodiment, in which:

Figure 1 is the system structure and main components of the embodiment.

Figure 2 is the procedure for finding at least N carparks that are near to a target location.

Figure 3 is the main GUI of the embodiment.

Figure 4 is another GUI showing a navigation button.

Figure 5 is the GUI that shows more complete information of a selected carpark.

Figure 6 shows a procedure for finding the cheapest carpark(s).

Description of an Embodiment of the invention

The mobile app system consists of a cloud based backend database (1-1) and application servers (1-2), the mobile phones with the app's front-end program (1-4) linked to the backend servers through the network (1-3). One of the main features of the mobile app system is to display key parking information of the carparks near to the target location together on one single screen, to help users to easily find the carparks that better fit their parking needs.

The backend database is for storing parking information and application related data. The backend servers are used to get data from parking data sources (1-5), to communicate with the database for getting and storing data, to perform application related calculations and functions, to communicate with the mobile phones, and etc.

The main functions of the app front-end program running on mobile phones include, but is not limited to: to receive user input data, to get the geo-coordinates of the mobile phones, to perform necessary front-end computations and functions, to communicate with backend servers for transmitting and receiving data, and to display GUIs on the mobile phone screen.

The parking information of about 3,000 carparks is stored in the backend database. The parking information stored for each carpark includes:

(A) car park code, name and short-name, address, geo-coordinates and etc.

(B) the total number of parking lots and the current number of available parking lots, the parking lot available levels for each applicable vehicle type;

(C) the parking tariff including all charge rates for every time zones in different day types for each vehicle type.

The system setup parameters related to the application, such as d_0=1km, a%=10%, b%=60%, and etc. are also stored in the database.

Figure 3 presents the main GUI on a mobile phone, where

- the round button with a magnifier image (3-7) is the search button. After clicking the search button, an input field will appear, the user can input data for searching a target location. The data input could be a building/carpark name, address, or postal code. During the data input, the system will display a list of building/carparks that are matched or partially matched to the data input, the more the characters are input, the lesser the number of carparks are in the list.
- the round button with a vehicle image (3-8) is the vehicle type selection button. The default vehicle type is car, when user clicks on the button, the image on the button will be changed to another vehicle type, in the order of car → motorcycle → heavy vehicle → container → car, and the parking information of all the carparks on the GUI will be changed correspondingly for the selected vehicle type.

- pin-shaped icon is for the carpark icon. The pin foot (3-1) is used for marking the carpark's location on the map. The pin head (3-2) is used for indicating the parking lots available levels with different colors (red color indicates the low availability (available level ≤ a%); green color shows the high availability (available level > b%); orange color means the middle availability (a% ≤ available level ≤ b%); blue color is used to indicate that there is lots available data but the lots available level cannot be worked out due to a lack of the total number of lots; black color means that the system cannot get the available lots data.
- the text (3-3) is the carpark/building short name that is displayed near to and above the carpark icon.
- the current parking fee charge rate (3-4) and the end time of the charge rate (3-5) are displayed below the carpark name and above the carpark icon.
- the sign (3-6) is used for marking the cheapest carpark(s) among the carparks shown on the map. The cheapest is defined as the lowest parking fee for the first 1 hour of parked time, starting from the current time, if the comparison condition is applicable. Otherwise there will be no cheapest carpark shown.

Once a user inputs/searches/selects a target location, the system will firstly find all the carparks near to the target location by using the distance condition of $d \leq d_i$ (refer to the procedure in the Figure 2). Secondly, it retrieves the 5 key parking data for each of the carparks near to the target location from the database. Thirdly, it finds the cheapest car park(s) (refer to the procedure in Figure 6), then finally, it displays the key parking data of all the carparks together with the cheapest carpark sign (if applicable) on the single screen (Figure 3).

The user can easily compare the key parking data of the carparks near to the target location, and quickly choose a car park that is better fitting to his/her parking needs.

Once he/she decides to go to the chosen carpark, the user can click on the carpark icon, and see the following parking information (Figure 4):

- the round button (4-1) that is used for seeing more parking information,

- the carpark/building's full name (4-2),
- the current lots available level presented in color with the number of available lots over the number of the total lots (4-3),
- the navigation button (4-4) with car park address and the navigation symbol.

In the case that the user wants to know how to reach the selected carpark, he/she can click on the navigation button (4-4) to switch to the navigation screen.

In the case that the user would like to get more complete parking information of a selected carpark, the user can click on the button (4-1), the Figure 5 will appear, where the following information are shown:

- the carpark/building's full name (5-1),
- the number of available parking lots over the number of the total lots (5-2),
- the navigation button showing the car park address and navigation arrow (5-3),
- the icons of the vehicle types (5-4) allowed to park in the carpark (the icon marked with black color is the selected vehicle type), user can change the selected vehicle type by clicking on the desired vehicle type icon,
- the name of the vehicle type selected (5-5),
- the parking charging tariff (5-6) including the charge rates in every time zones for different day types for the selected vehicle type.

Claims

The claims defining the invention are as follows

1. A method and GUI design that defines and presents 5 key parking data (carpark name, geo-coordinates, parking lot availability, current parking fee charge rate, and the end time of the charge rate) together on a single display screen for all the carparks near to a target location being input by a user.
2. A method and GUI design according to claim 1, wherein the key parking data of a carpark is presented by using a carpark icon together with texts/graphic surrounding the icon on a geographic map.
3. A method and GUI design according to claim 1 or claim 2, wherein at least 3 of the 5 key parking data is presented together on a single display screen for the carparks on the screen.
4. A method and GUI design according to claim 1 or claim 2 or claim 3, that includes a procedure that is used for finding the carparks near to a target location within a search area limitation and that ensures it finds at least N (N is a pre-programmed integer) carparks near to the target location with reduced computations.
5. A method and GUI design to find and present the cheapest carpark(s) among the carparks by using a cheapest carpark sign together with the key parking data on a single display screen.