TimedNavigation - A Time-based Navigation Approach

Janko Timmermann
OFFIS – Institute for Information Technology
Oldenburg, Germany
timmermann@offis.de

Benjamin Poppinga
OFFIS – Institute for Information Technology
Oldenburg, Germany
poppinga@offis.de

Martin Pielot
Telefonica Research
Barcelona, Spain
pielot@tid.es

Wilko Heuten
OFFIS – Institute for Information Technology
Oldenburg, Germany
heuten@offis.de

Susanne Boll
University of Oldenburg
Oldenburg, Germany
susanne.boll@uni-oldenburg.de

MobileHCI '13, Aug 27-30 2013, Munich, Germany
ACM 978-1-4503-2273-7/13/08.
http://dx.doi.org/10.1145/2493190.2494656

Abstract
Travelers sometimes need to reach a destination in a given amount of time. However, today's navigation systems try to route users to the destination as fast as possible. In this paper, we present the concept of time-based pedestrian navigation. We use a map on a smartphone that highlights streets depending on whether they will lead to a destination in time. Our map also allows the users to choose between route alternatives during the walk.

Author Keywords
Navigation; Abstract Maps; Time

ACM Classification Keywords
H.5.2 [User Interfaces]: User-centered design

Introduction
When navigating, people tend to reach their destination using the shortest path that takes least time, but sometimes people just desire to be walking for a certain time. One example could be to explore a city for two hours before heading back to the railway station.

Timekeeping is an important factor e.g. to ensure not being late for an appointment. Not following the orders of a conventional navigation system may lead to taking too long ways and missing time limits, because taking the wrong turn in some environments like forests can have a big impact on the needed walking time. We therefore propose an approach which enables users to use a specified time period efficiently and allows them to take a route leading to the destination just in time. Our approach also supports serendipitous exploration through dynamic navigation. This differs from the normal shortest or quickest path approaches and encourages people to think about exploring an area rather than taking the same repetitive route.

We developed a mobile application called *TimedNavigation* that allows users to specify a destination and the desired travel time. Whenever the traveler reaches a junction, the app calculates for each turn whether it will be possible to arrive at the destination in time or if the user will be too early or too late. Each possible route is colored to communicate the results to the user (yellow = on time, green = too fast, red = too slow). Thus, the application enables the user to reach his or her destination in time, without being too late and avoiding being to early.

Related Work

In our approach, we use abstract maps for displaying the navigation information. Radoczky and Gartner [3, 2] tested several types of abstracted maps. They found that most participants performed well with all tested maps. Their results also indicated that on an unknown path the map representation could influence the generation of the users' mental map of the area. Ekman et al. [1] showed that even an arrow can be sufficient to guide user to a specific destination. Nevertheless, they reported that users were not satisfied with that solution and like context such as maps which, on the other hand, can confuse them.

The related work also shows other navigation approaches, which do not propose a concrete path like our approach. An example is the work of Robinson et al. [4], who introduced a concept where users are guided in general direction of their destination using vibration. They tried to encourage the users to a more exploratory type of navigation by varying this feedback based on the potential of various routes.

Concept

We wanted to support users in navigational tasks with a time-based background. This means reaching a destination at a given time. This includes not only navigating from one point to another, but also walking round trips, thus reaching the start point again at a specific time. Possible scenarios include reaching a bus stop at the departure time, reaching a meeting in time, or use a lunch break for a short walk. Even health related scenarios can be thought of, because the app can support users to walk more, even in situations where they normally would not walk because they could be late for their next appointment. In these scenarios, time constraints can be critically important. Current navigation solutions do not regard time constraints in that way, but propose e.g. the shortest and fastest way.

A possible solution would be to calculate one possible route that would allow to reach the destination in time. This would be strict and is probably not a sufficient solution for the mentioned scenarios, because it does not give the user the feeling of usable spare time. In an early requirement analysis we also found that users like to choose routes by themselves if they seem to be more attractive to them than the planned route. We therefore decided to propose a system which not only enables users to reach their destination in time but also gives them the freedom to choose the way on their own dynamically, without previous planning. For this purpose, an easy to understand visualization is needed which enables the user to easily check if s/he can take a preferred way without being late. We propose to use a smartphone application using an abstract map with strong contrasts and an easy to understand color scheme to encode this information.

Design of the Application

We created a smartphone app which we called *TimedNavigation*. The purpose of the *TimedNavigation* app is to navigate users to a specific destination at a specific time using a route chosen by the user him- or herself.

After entering the destination and the desired arrival time, the app enters the navigation mode. In the navigation mode, only crossings are regarded, because they are the decision points for changing a route. The app automatically chooses the nearest crossing and displays it if the user is at most 50 meters away, assuming s/he is already standing there. If a crossing is found, the application proposes one route to the destination for every street connected to that crossing. For example, at a crossing with four connected streets, four routes (*route1, route2, route3, route4*) are created, see *Figure 1*. These routes always lead to the next crossing from where they continue with the minimum walking distance to the destination without turning back. In most cases this might not be exactly the route the user actually wants to walk, so this routing is performed again at every crossing the user approaches.

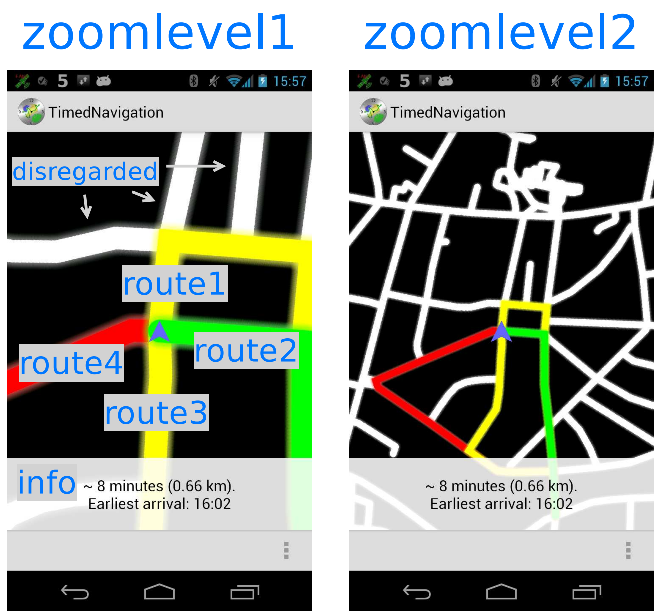

Figure 1: *TimedNavigation* app with two different zoom levels and annotations (not visible in the application) on the left screenshot.

To keep the map simple and abstract, there is no distinction between street types like trails or highways. Only time information is used to vary the colors of the streets. We decided to use traffic light colors to encode the time information. Further, we used white as neutral color for streets which are not directly connected to the current crossing and are not used in any of the computed routes, see *disregarded* in *Figure 1*. Streets are colored *green* if the route leads more than four minutes too early to the destination, yellow if less than four minutes and red if the user will be too late when taking that street. *Figure 1* shows an example for every encoding.

In this way, the common order of the colors, like at traffic lights, is kept and the users usually see green streets at the beginning of the navigation, later also yellow and red ones.

While the colors are meant to give a quick overview, we also implemented a small information (*info*) area at the bottom of the screen, see *Figure 1*. This includes the duration, the distance, and the arrival time. If the user points towards a street in the real world, this information is updated with exact data about the shortest possible route starting with this street. The map is always oriented with the help of the compass so it always matches the environment.

The app has two zoom modes (*zoomlevel1* and *zoomlevel2* in *Figure 1*). The first mode is very close to the current crossing. In this mode, normally only the current crossing and very close following crossings are displayed. It is the standard mode and meant to focus the user on the current crossing, thus only giving the necessary information. The second mode gives an overview of the surroundings of the current crossing, which is nevertheless also in the center of this view. This given overview is not necessary for this navigation concept, but was demanded by users during early requirement analyses. The first mode uses a resolution of 0.3 meters per pixel while the overview mode uses a resolution of 1 meter per pixel.

The application relies on the map data of OpenStreetMap. It is not using any of the pre-rendered maps the OpenStreetMap Community is offering, but dynamically renders the map itself. Additionally, the data from OpenStreetMap is used to dynamically create the different routes, since using an online routing service would be much slower because of the high amount of routing requests.

Future work

We already did a first user study and compared our application to the Google Maps app, which showed promising results and revealed that participants were significantly more punctual when using our app. Further to extending this study, we plan to use such type of navigation also in other contexts and use e.g. the attractiveness or the difficulty level of a route.

References

[1] Ekman, I., and Lankoski, P. What should it do?: key isssues in navigation interface design for small screen devices. In *CHI'02 extended abstracts on Human factors in computing systems*, ACM (2002), 622–623.

[2] Gartner, G., and Radoczky, V. Schematic vs. topographic maps in pedestrian navigation: How much map detail is necessary to support wayfinding (2005).

[3] Radoczky, V., and Gartner, G. Extent and effectiveness of map abstraction for communicating routes in a lbs (2005).

[4] Robinson, S., Jones, M., Eslambolchilar, P., Murray-Smith, R., and Lindborg, M. I did it my way: moving away from the tyranny of turn-by-turn pedestrian navigation. In *Proceedings of the 12th international conference on Human computer interaction with mobile devices and services*, ACM (2010), 341–344.