# Line navigation: AR indoor navigation


**1st Author Name**
He Gong

**2nd Author Name**
Junbo Hu

**3rd Author Name**
Ziqiu Wu



## ABSTRACT

UPDATED—11 December 2018. This sample paper describes the formatting requirements for SIGCHI conference proceedings, and offers recommendations on writing for the worldwide SIGCHI readership. Please review this document even if you have submitted to SIGCHI conferences before, as some format details have changed relative to previous years. Abstracts should be about 150 words and are required.



### Author Keywords

Authors' choice; of terms; separated; by semicolons; commas, within terms only; this section is required.

### ACM Classification Keywords

H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous; See http://acm.org/about/class/1998 for the full list of ACM classifiers. This section is required.


## INTRODUCTION

In this paper, we will introduce an AR indoor navigation application, which utilize computer vision technology and AR technology.  To accomplish this, we analyzed the current indoor positioning technologies and summarize the advantages and disadvantages of them, then we begin to design our own application which have high accuracy, low cost. For implement our design, we designed three low-fidelity prototype and chose two of them for working prototype implementation.

## PROBLEM STATEMENT

In this part, I will introduce the background knowledge of indoor navigation problem, compare existing solutions and summarize their advantages and disadvantages, and finally, I will briefly introduce the methodology of our project.

### Indoor navigation problem

First of all, why do we need indoor navigation? There are two reasons for this: First, people spend more time indoors than outside. One an average day in Beijing or Hong Kong, you might walk several miles without ever going outdoors. This is possible thanks to the extensive system of interconnected shopping malls, office towers, covered elevated walkways and subway stations. Second, large buildings such as airports, hospitals, shopping centers, etc., often have complicated internal structures, so people may not be able to recognize their position or direction in these mazes and lose their way easily.

Since indoor navigation is needed by people, why is there no such software on the market? There are still two reasons for this: First, in the absence of GPS signal support, it is difficult to find a perfect replacement; second, it takes a lot of manpower and material resources to present complex indoor structures in 2D/3D forms. Simply speaking, the navigation system is composed of a positioning system and a map. Despite drawing maps is out of the scope of this paper, it is not negligible. Even if we have mature indoor positioning technology, we still need detailed indoor maps to let the navigation system work.

### Previous work

Most of the existing indoor navigation systems in the market cannot escape the mode I mentioned above. In this part, I will show you some indoor positioning technology.

#### Wifi-Positioning Technology

The working principle is to calculate the relative distance between the location and routers through the received signal strength of routers. After knowing the precise location of routers, we can obtain the absolute positions of users. It has some disadvantages, like low anti-interference ability and relatively low accuracy.

#### Bluetooth-Positioning Technology

Its working principle is similar to Wi-Fi technology. It is more accurate than Wifi-positioning technology with higher cost, and its anti-interference is bad.

#### Inertial navigation Technology

Its working principle is similar to Wi-Fi technology. It is more accurate than Wifi-positioning technology with higher cost, and its anti-interference is bad.

#### Computer vision Technology

It is an ideal method. But we have to use massive data to train the system for making it usable in different scenes. And We still need to draw maps for different buildings.

#### Magnetic-Positioning Technology

It requires no additional hardware and it can provide high accuracy. But you have configured your application, like manually doing the map setting, gathering magnetic information on the spot. If the allocation in the building changes, the system will fail.

### Methodology

After discussing other technologies, I'll give the proposal for our project. There are three major objectives which promote our design: low cost, precision, and plug and play. The main difference between our system and other systems discussed above is that we do not need to draw any indoor maps for buildings, which means we can use the application anywhere. The technical level is based on computer vision technology, more precisely based on the Augmented Reality drawing supported by ARkit. By using computer vision, the software can convert the surrounding environment to multiple 3D points which are all space coordinates. As we move we can draw a 3D line composed by multiple space coordinates. And users can share their path with each other. In this software, the user role will be converted, the user not only enjoys the service but also provides the service to others. This concept of cooperation gives the software the possibility of socializing.

## CONTEXT INQUIRY

We inquired some questions to some students in our school. They holds different viewpoints. We detailed our app and asked them whether they would use this app. Someone supported it thoroughly, "Yes, this is useful because there are no similar apps currently". Someone thought it is useful, "Yes, I always lose my way in the building". Someone didn't want to use it, "My phone doesn't support ARCore".

According to the answers, we build a sequence model, as shown in figure 1.

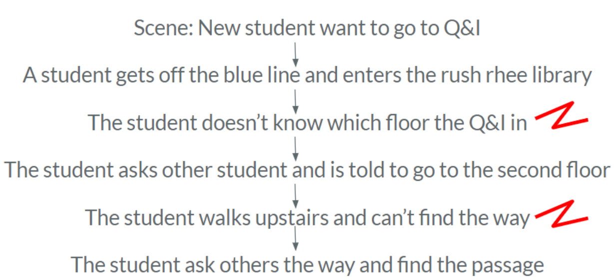


**Figure 1.Sequence Model**

## STORYBOARD

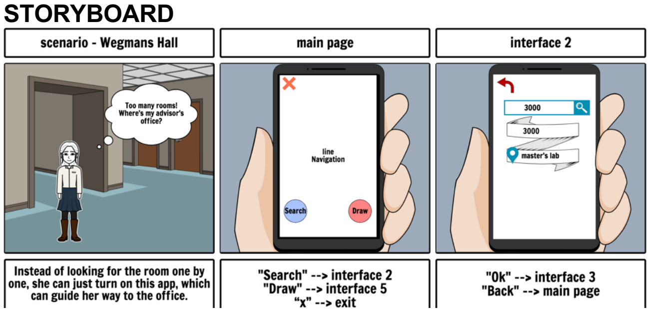


**Figure 2. A girl wants to drop by her advisor's office. She enters into the Wegmans Hall but doesn't know how to get there.**

Illustrate one of the main scenarios: A girl wants to drop by her advisor's office. She enters into the Wegmans Hall but doesn't know how to get there. Instead of looking for the room one by one, she can just turn on Line Navigation, which can guide her way to the office, as shown in Figure 2 (a).

When she turns on this app, the first interface offers her two options, as shown in Figure 2(b). One is for searching the route line. The other one is for drawing the route line.

Now, she can touch the "search" icon and turn to interface 2. The location (or keywords) she wants to go can be entered in the input field. The other recommended or nearby locations (according to GPS) are listed below, as shown in Figure 2 (c).

After she chooses the desired place to go, the AR route line will be shown up in the air through the camera, which can guide her to go upstairs and find her advisor's office, as shown in Figure 3.

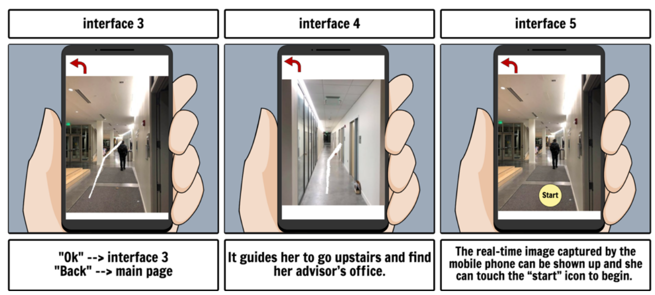


**Figure 3. The girl is leading by an AR line**

There are many other useful scenarios such as in the hospital, the airport or the shopping mall. The more usage in the field, the more routes are likely to be shown up.

Back to the main page, if she wants to create a new route, she can touch the "draw" icon, which allows users to draw their own route in the air.

The real-time image captured by the mobile phone can be shown up and she can touch the "start" icon to begin, as shown in Figure 2(c). When she is walking, the line continues along the path. When she finishes, touch the "finish" icon and a window pops out allowing users to name the route line and then save it locally or upload to the Internet, as shown in Figure 4.

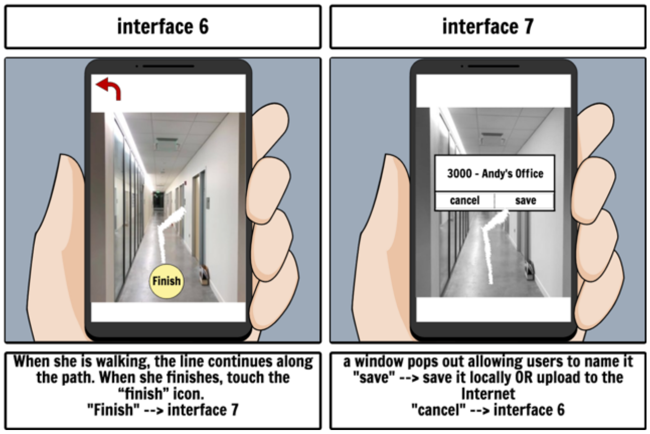


**Figure 4. Finish the route and save it**

## IMPLEMENTATION

Our app is based on ARCore and android system. We can split the development of app into three parts according to the roles of users. We need users to create the route so we call them "Route Maker". The users who use the route are called "Route User". The first part is that route maker starts the app, records the route, ends the recording and saves the route into a file. The second part is that the route is uploaded by the route maker into our server and downloaded by the route user from our server. The third part is that the route user searches the route, loads the route and navigate with it.

For the first part, when user starts to record the route, the camera will automatically take a picture to record the original perspective of the route and set the position of beginning point as the zero point of the world coordinate. When the route maker walks, our app will get the projection matrix and other matrices via ARCore library to calculate the position of the user. When the user has moved more than a distance, we will add his relative world coordinate to the route and use OpenGL ES 2.0 to render the route as a white line on the screen. Because too many points will take up too much room, we will reduce the close points and make it a straight polyline.

For the second part, we will send the route to our server with UDP. Our server will split the route and make index with the labels of routes. When user sends a search request to the server, the server will return a appropriate route to the user.

For the third part, the user will request a route from the server, load it and display the original photo. When he find the beginning point, he can start the navigation and follow the AR polyline.

## EVALUATION

We may evaluate it in different scenarios. Let's take the Wegmans Hall's as an example. We will invite students/ faculties or citizens (of different ages) that have never been to Wegmans Hall before (such as students from medical center), and randomly divide them into two groups. Let group one uses the Line Navigation first while group two use similar navigation app such as Google Map first. Each person of two groups should take at least one pair of tasks, one task using this app while the other task using Google Map to reach a different location.

| Group | | | |
|---|---|---|---|
| 1 | 3000 | 1400 | 3001 |
| 2 | 3000 | 1400 | 3001 |

**Table 1.**

We may set them start at the same place (such as the gate of Wegmans Hall), one by one. We collect the data and figure out which group can find the location more accurate (whether find the correct room or not) and faster (record the time used). Then, use t-statistic to analyze whether the factor of app has significant impact on the task. Also, we may design questionnaires to users to get the feedback such as the experience feelings during using the apps.

## LOW-FIDELITY PROTOTYPE

### MOTIVATION

While mobile navigation is omnipresent outdoors, it is not inside buildings because of the normally non-existent GPS reception. Wi-Fi is often used in this case to create a so-called "indoor GPS". However, its accuracy is 5-15 meters and iOS devices are excluded from client-based positioning. Beacons are the alternative, which have accuracy of 1-3 meters, while interference can occur when beacons are installed in a space with lots of Wi-Fi signals.

In order to make a high accuracy, low cost and easy plug-in system, we decide to use computer vision technology, with Augmented Reality lines guiding the ways. This app is for indoor places such as a hall, a library, parking garage and subway transit system. The more people use the app, create the routes and voted for the popular routes, the more well-established the system will be.

We build a paper prototype based on three main scenarios.

The first one is to record a new route. Imagine you are a freshman to University of Rochester and someone is guiding you on campus tour. During the process, you can use Line Navigation to record some routes to key places such as Bai's office. After choosing 'Record Route' button, the real-time image captured by camera is shown up on the screen. Then, you can touch the 'Start Record' to start recording your routes. As soon as you do so, the icon turns to 'Finish', which can be taped whenever you want to end this route. When she is walking, the line continues along the path. There is a chance for you to make sure whether this is the route you want to create. The two images of starting point and ending point will be posted on screen and a text field above them allows you to type the name of the building, and two text fields under images allow you to type the place you start and finish. Last, touch 'SUBMIT' and a Route Number is returned.

It's especially useful for UR students or faculties to use Line Navigation to go through the tunnel because of its complex inner structure.

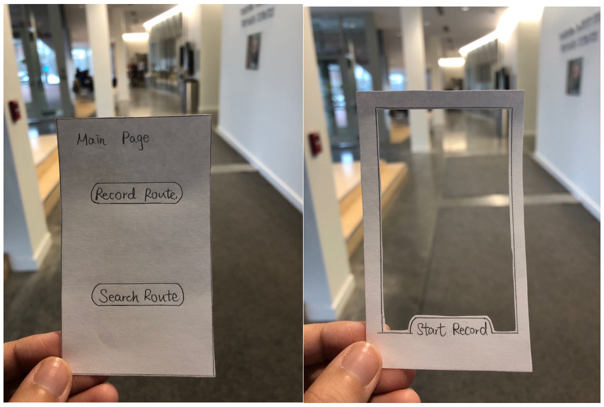


**Figure 5.**

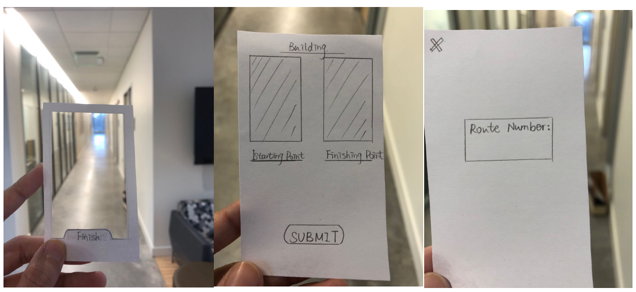


**Figure 6.**

The second is to find the place users want to go. For example, a girl wants to drop by Bai's office. She enters into the Wegmans Hall but does not know how to get there. Instead of looking for the room one by one, she can just turn on Line Navigation, which can guide her way to the office. Now, she can touch the 'Search Route' icon and the location (or keywords) she wants to go can be entered in the input field. After she enters the information of building, starting point and destination, type 'Search'. Then a translucency image of the starting point will be superimposed on the camera image, allowing her to make sure this is the right place to start guiding. 'Start', and an AR route line will be shown up in the air through the camera, which can guide her to go upstairs and find her advisors office. Once click 'Finish', interface goes back to the main page.

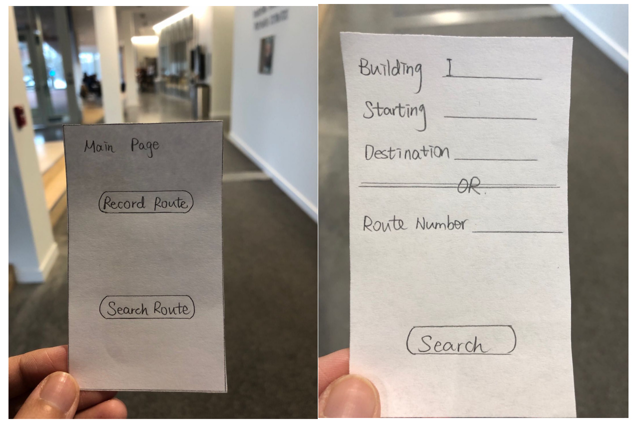


**Figure 7.**

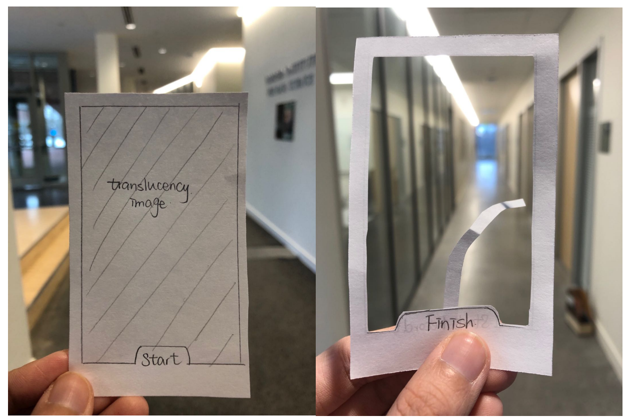


**Figure 8.**

To achieve more communications between users, Line Navigation allows user to share his creation of route with other user. So here comes the third scenario, following a route shared by a friend. Recall that in the first scenario, there is a route number returned to you when you create a new route. You can keep this number for yourself or send to your friends. For example, a librarian has created a detailed tour route of Rush Rhees Library, which can be followed by students by entering route number. We converge this function with the second one as they both belong to 'Search Route'. However, there is a big difference between them. In scenario 2, the goal of a user is to find the right place. So the optimal route with the shortest distance will be returned to user. In this scenario, the goal is to follow a tour guide. So the duplicated or crossed line should not be removed.

## WORKING PROTOTYPE

There are two working scenarios we successfully implemented in this working prototype. They are drawing routes and using routes respectively. And we packaged these two scenarios in one mobile application.

### Design

We want to design an mobile application which can allow people to draw their routes, in another word, record the path they have taken. Because we are designing indoor navigation application, we can't use GPS signal to achieve positioning. In addition, we do not want to install some other devices that can transmit signals to assist us in positioning. So we think about using Computer vision technology to identify the surrounding environment and generate a point cloud for navigation. This technology can transfer surrounding environments into point clouds which we can recognize as a 3 dimension map. Then we can add extra points to the cloud as the path we walked, just like we used the pen to draw the route on a paper map. When we finished drawing the route, we can share our route with other people to help them find their destination.

The computer vision technology is for drawing the route, and the AR technology is for reproducing the route. The reason we use the AR technology is that it can present virtual elements in real environment by some devices. There are already a lot of navigation applications take advantage of this benefit of AR.

The above is the basic design of our software. Below I will explain the design of each scenario separately. In the first working prototype, there are a few functions. First, users can "draw" their routes by simply tapping the button on the mobile phone screen and they can also end the recording by press the virtual button again. The second function is naming routes. After drawing a route, the user can name the route he or she drawn for other people's convenience. And the last function is uploading the route or deleting the route. In the second working prototype, people will find a route and use the route for their navigation. So we design an interface for users to search the route. The users have to enter a pair of start point and destination. When the user chose a route, the virtual signs will show up on the screen and he or she can follow it. I will explain how this application can determine the starting point position in next part.

**Implementation**

We implemented our prototypes on an Android mobile phone which supported ARCore application. The reason we chose the mobile phone as our application platform is that it is easy to implement and the device is easy to get.

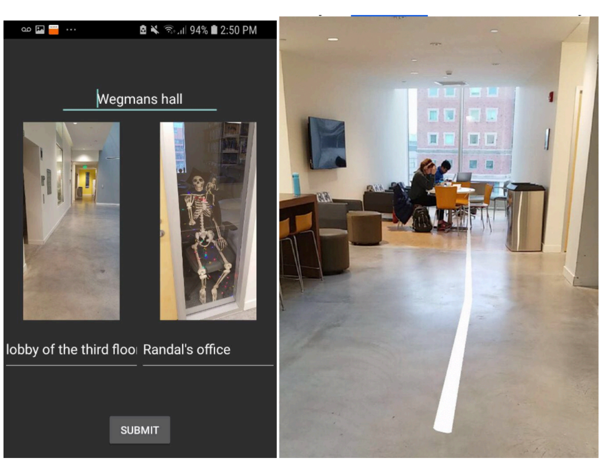


**Figure 9. one interface picture and a snapshot of the screen**

The implementation process is tedious and nothing special. I will focus on some findings in our implementation process. The first major problem of the implementation is the recognition of the starting point. The route can only work when the user is at the location which is exactly the start point of the drawer. Not only the position, but also the direction. The user has to stand at the same position and face the same direction as the drawer did. The current solution of this problem is to use a picture for determining direction. Before user starting drawing the route, he or she has to take a picture which represents his walking direction. And the application will send the route and the picture together to other users. Then before other users using this route, he or she will find the starting point based on this picture and the word description which is also input by the drawer. We can let the user to compare the picture with the surrounding environments themselves, or we can use the mobile phone camera to automatically recognize the picture. This solution may be a little bit complicated and inconvenient and we will try to find other better solutions to improve the performance in the future.

For better user experience, we also did a lot improvement. Users don't have to continuously tapping the virtual button, he or she only has to tap the button at the starting point and the ending point. Second, we smooth the route while recording it. We modified the altitude of all the points to a specific number. Second, we removed redundant points. By doing this part, we not only make the route more concise but also reduced the storage area. When we doing the recoding, we use a 3D point to represent the user position. If we only tracks the position of these points, the route is messy and it also requires much more storage space. So we design an algorithm to reduce some noisy points. For example, if the user walks in a relatively straight line, we will delete all points except the start point and the end point. By this method, the navigation routes in our application are composed by line segments. The details are as follows: suppose the current position is a, and after the user walk a while he reached position b, which is away from a with a threshold distance. And then the user reached the point c. At this time, we calculate the angle between segments ab and bc. If the angle is less than a threshold number, we believe that we think that their three points are in a straight line.

## MAIN FINDINGS

The biggest discovery is that this project is not as easy as it seems. The virtual navigation lines sometimes drifts. And we are still trying to figure it out.

## IMPLICATION

By doing this application, we have gained a lot of sentiments. First of all, the idea is very important, and secondly, it is the means and tools to realize it. When you want to design a software, your inspiration and ideas should be close to your life. As long as you are a user of this software you can design a useful software. This is also in line with user-centered design theorem. Second, it’s easy to think but hard to implement. When I first thought of the idea of this software, I think it’s easy to implement and I did not expect to encounter so many problems in the actual implementation. No one can come up with a perfect solution at the very beginning. A lot of improvements happens when you start to construct your project. So the important thing is not how good your idea is, but how early you are to implement your project. Another important thing is self-confidence. Only if you think your product is good, can you make other people think that your product is good.

## FUTURE WORK

First, we need to find a better way for positioning. The current solution is complicated and not that precise, especially for the automatic identification. Second, we may choose to expand the application area of this application. Since we can't solve the problem of indoor precise positioning now, we can just use this application without positioning or fixed the start position of the navigation. For example, we can combine this application with escape routes. We can attach a QR code of an escape route on an escape indicator. Then if something happens, people can use this to find the best escape route. Once this application is popular, we can also use users’ route selecting data to mine some interesting patterns behind. And also, we may try to implement this application on other platforms such as google glass. Another feature of our project is the sociability. Because our project is based on users. Users not only use this application but also implement and construct this application. So I think the sociability of this project is indispensable.

## CONCLUSION

In this paper, we introduce Line navigation, and find that it can reduce the infrastructures for indoor positioning, while still providing easy and precise navigations solutions. Despite the current positioning solution is not perfect, it is already useful for short distance indoor navigation. in evaluation tests, we sometimes find the virtual routes may drift in some situation and we will work on this part in the future. And this application has the potential to be used in other areas in the future.

**Reference:**


1. Mo ̈ller, A., Kray, C., Roalter, L., Diewald, S., and Kranz, M. Tool support for prototyping interfaces for vision-based indoor navigation. In Workshop on Mobile Vision and HCI (MobiVis) on MobileHCI 2012 (2012).
2. Mulloni, A., Seichter, H., Du ̈nser, A., Baudisch, P., and Schmalstieg, D. 360 ◦ panoramic overviews for location-based services. In Proc. of the SIGCHI Conf. on Human Factors in Computing Systems (CHI), ACM (2012), 2565–2568.