Smart Toy Car Localization and Navigation using Projected Light

Mingming Fan
Computer Science Dept.,
University of Toronto
mfan@cs.toronto.edu

Qiong Liu
FXPAL
liu@ fxpal.com

Shang Ma
EECS
UC Irvine
shangm@uci.edu

Patrick Chiu
FXPAL
chiu@fxpal.com

***Abstract*—In this paper, we present the design and implementation of toy car localization and navigation system, which enables toy cars and "passengers" to learn and exchange their fine-grained locations in an indoor environment with the help of the projected light based localization technique. The projected light consists of a sequence of gray code images which assigns each pixel in the projection area a unique gray code to distinguish their coordination. The light sensors installed on a toy cars and a potential "passenger" receive the light streams from the projected light, based on which their locations are inferred. The toy car then utilizes A* algorithm to plan a route based on its location, its orientation, the target's location and the map of "roads". The fast speed of projected light based localization technique enables the toy car to adjust its own orientation while "driving" and keep itself on "roads". The toy car system demonstrates that the localization technique and the client-server architecture can benefit similar applications that require fine-grained location information of multiple objects simultaneously.**

Keywords-toy car; indoor localization; navigation projected light; projector; smart object; multi-device interaction.

I. Introduction

Location is an important information to perform context aware computing. For example, it can help cars localize themselves and drive on roads. It may also be used in smart homes to provide people with location related services. However, acquiring location information, especially in an indoor environment, is still challenging today. The result of a recent indoor localization competition [6] reveals that designing appropriate applications, improving localization accuracy and reducing the deployment overhead are among the challenges for the state-of-art localization techniques.

In this paper, we introduce a smart toy car localization and navigation system that is based on the projected light localization technique that we have designed [3]. Even though big driverless cars have been enabled by Google and other companies, those technologies are not going to work for toy car navigation in an indoor environment, partially due to the lack of GPS signals and the coarse localization accuracy enabled by GPS. To tackle this challenge, we use projected light, which consists of a sequence of coded images and assigns each pixel in the projection area a unique code to distinguish their coordination. With the projected light based localization technique, our system can localize multiple devices simultaneously and allow them to exchange their fine-grained location information in an indoor environment.

Our toy car localization and navigation system utilizes an overhead projector which is mounted on the ceiling and projects a specially designed beam of light to the floor. Any pixel in the projection area is illuminated by a unique sequence of gray code that distinguishes its location from all other pixels. Each one of the two light sensors installed on a toy car receive a stream of light intensity readings, from which their locations can be computed. Based on the two light sensors' location, the toy car's position and orientation can be inferred as well. A similar localization method is used to get the location of a "passenger". When the "passenger" sends a "picking me up" request to the server, the server will compute and forward the person's location to a nearby available toy car. Once having the knowledge of the location of the target "person" and a map of all available "roads", the toy car will compute an optimal route and navigate to the "passenger".

The projected light based localization technique and the client-server architecture can accomplish the following three challenging tasks simultaneously: 1) exchanging high-precision location information among multiple devices (Toy car and the simulated "passenger") in real time; 2) providing real-time navigation guidance; 3) localizing multiple toy cars at the same time. Additionally, the system also demonstrates the following advantages of the projected light based localization technique. First, the projected light beam for localization is perceived as a constant-brightness illumination light source and does not change the environment appearance obviously. Second, no calibration is required. Lastly, because no camera is used, the privacy of any present people can be preserved. We envision our system serve as an initial exploration example to demonstrate that projected light localization can localize locations of objects and people presented in an indoor environment that can have a potential to benefit other applications.

II. Related work

Despite lots of efforts have been devoted into indoor localization from both academia and industry, indoor localization is still a challenging topic in terms of both localization accuracy and potential using scenarios. A recent practical indoor localization techniques competition reveals that indoor localization problem has not yet solved [6]. The best performance achieved in the competition is 0.72 meters localization error. Applications that require room-level or even meter-level accuracy may be powered by such technologies. However, for the tasks, such as navigating in a shopping mall consisting of lots of individual shops and the toy car navigation, require more fine-grained location information so as to determine which shop the user is current at and guide the car drive only on the designated "roads" but not hit any "buildings" off the "roads". Our projected light based localization technique can distinguish locations at the projected pixel level (~2 mm granularity if the projector is mounted on the ceiling of a standard office room, which has height of 2.7-3 meters typically).

In terms of guiding robots navigating in an indoor environment, paper visual tags have been investigated to inform robots the content and position of objects. Magic Card [9] designed different paper cards to represent different tasks. By placing cards at different locations on the floor, robots can learn their locations and come to perform the related tasks (*e.g.,* vacuuming). Similar concept was studied to teach a robot the locations and content of different ingredients so that it can grab them and cook accordingly to a receipt [8]. Although using paper tags to teach robots the content and location of a task is a fun way of human-robot interaction, informing a robot of its own location and locations of other surrounding objects can be done by using our projected light based technique and without visually obtrusive tags.

By capturing a set of key images using embedded camera and ordering them topologically, a robot can use them as the knowledge to compare with while navigating in the same environment [2]. In contrast, our toy car localization system does not require any prior knowledge of the environment, and the locations of itself and the target are computed in real time.

By projecting a sequence of gray code images to an arbitrary size and oriented surface whose four corners are installed with four light sensors, Lee et al. [5] designed a technology that can calibrate the projection onto any shape surface. A similar technique was used to recognize the boundary of a TV so that a projector can extend the game content to be projected beyond the TV's screen in order to increase the immersion of the game experience [4]. With the projector installed on a vertical wall and projecting horizontally, Fan *et al.* [3] used a similar technology to author digital content to different paintings on a vertical wall by using a light sensor to discover all different paintings' positions on the wall in real time. In contrast, we explored how an overhead projector set-up can be used to localize multiple-device simultaneously and enable them to interact with each other (in our case, navigating from the car's current location to the target "passenger").

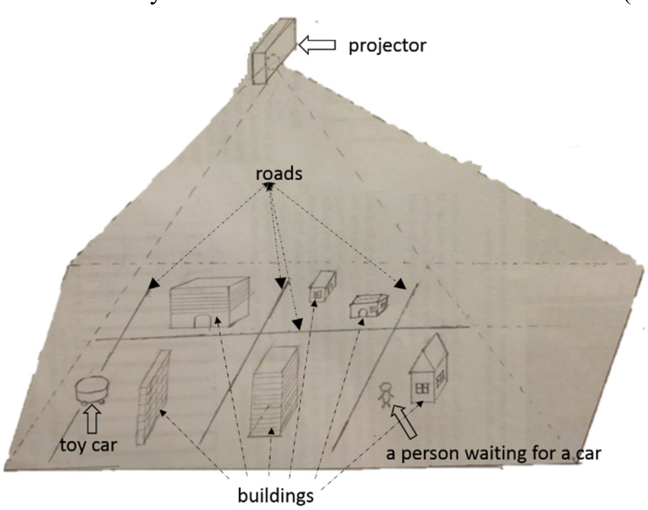

Figure 1. A sketch of the simulated environment for a toy car to pick up a "passenger". (Top) a projector projects the specially designed light beam to the floor and acts as an indoor GPS for the "car" and the "passenger".

III. TOY CAR LOCALIZATION AND NAVIGATION SYSTEM

A. System Overview

Figure 1 shows a sketch of the simulated toy car localization system. A projector is mounted on the ceiling of an office room and projects a beam of light to the floor. The light beam consists of a pre-defined sequence of gray code images, which are designed in such a way that each pixel of the projection area is illuminated by a unique sequence of gray codes. In Figure 1, both the toy car and the "person" are equipped with light sensors, which sense the light intensity. By decoding the light intensity stream received, the toy car and the "person" can be localized in real time. To simulate a realistic setting, the car is only allowed to drive on the designated roads. Our task is when a "person" submits a pick-up request, the toy can plan the route and navigate to the person without "driving" off the roads. To finish this task, the toy car needs to know its own location and orientation, the location of the "person", and a map of available roads. Additionally, it needs to be able to adjust its own heading in real time so that it can always keep itself on roads.

B. Localization with Projected Light

The details of how a light sensor's location is detected can be found in our previous system [3]. Each toy car equips with two light sensors at two ends of the diameter of the toy car body. Thus, our system can also compute the orientation of the toy car based the two light sensors' location information. The position and orientation of the toy car can be computed using the following equation: $(\frac{x_A+x_B}{2}, \frac{y_A+y_B}{2})$, $\overrightarrow{(x_A-x_B, y_A-y_B)}$. (The coordinates of two light sensors A and B are: (x_A, y_A), (x_B, y_B)).

C. Recognition of Roads

To avoid collision and simulate the reality, cars are only allowed to drive on "roads". Figure 1 shows four roads where the toy car can drive on. In order to reason whether a car is driving on or off the road, the car needs to have the knowledge of the coordinate of all roads, which are considered as the "map" for the car. Our system recognizes all roads' trajectories using an interactive approach.

The details of the interactive approach is shown below. A person holds a light sensor and places it at one end of a road on the floor. The light sensor reads the light intensity and sends the sensor data stream to the server, where an app is running to receive the data and decodes the light sensor's current location. The app also records it as the start position of one road. The person gradually moves along that road until she reaches the other end. The app decodes all the positions and stores them as the trajectory of that road. Then she moves to another road and repeats the above procedure. The procedure ends when she finishes moving along all roads. Because our localization can decode about 80 positions per second, the person can move the light sensor while walking in normal speed.

D. Navigation

We leverage A* algorithm to plan the route from the toy car's current location (x_c, y_c) to where the waiting person stands (x_p, y_p) (Figure 2). Because we already get all the roads' trajectories using the method described in 3.3, it is easy to compute all the intersections of any two roads. The intersection *Inter(i , j)* of any two roads i and j becomes a searching node in the path planning. The path planning algorithm first navigates the toy car to the nearest intersection to its current location, and then uses A* algorithm to navigate to the next intersection. The process repeats until it reaches the closet intersection to the waiting person's position. The last path is from that last intersection to the projected position of (x_p, y_p) onto the nearest road: $(x_p{'}\ ,\ y_p{'})$. We define the cost function *Cost(X, Y)* of two intersections X and Y in A* algorithm using the following equation: $\begin{cases} distance\ between\ X\ and\ Y, & if\ directly\ connected \\ \infty, & if\ not\ directly\ connected \end{cases}$, and use Manhattan distance as the heuristic function to evaluate and choose the next intersection for the planned route until we reach the last intersection that is closest to the "passenger".

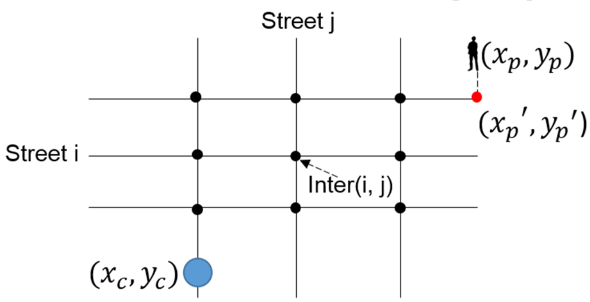

Figure 2. Illustration of the navigation map grid.

E. Implementation Details

We fully implemented the toy car localization and navigation system a deployed in a small office environment. The projector and the floor set ups were shown in Figure 3. The white dash lines in Figure 3 (right) highlighted the three of four boundaries of the projection area. Note that the projection area was illuminated by the projector's light beam and appeared to be slightly brighter than the surrounding area. There were four roads on the floor labeled with white adhesive tapes, where the toy car was allowed to drive on. Notice that the white tapes were not used for robot localization or navigation. They were labeled on the floor only for the purpose of giving visual feedback to human observers so that human observers know whether the toy car drives on road or not. A robot simulated a person who requested to be picked up. The trees and houses simulated the obstacles. The projector resolution was 912 * 1140 and the size of the projection area on the floor was about 3 m * 2 m. Therefore, the size of one pixel was about 2.2 mm * 1.7 mm.

We customized a commercial robot Bero [1] to build the "passenger" who requested to be picked up (Figure 4 left). A light sensor was added on its top side so as to receive projector light easily. A battery was attached to power it. The light sensor was also connected to a Bluetooth audio jack, where a Bluetooth audio transmitter was plugged in. We also customized a commercial robot SmartBot [7] to build the toy car (Figure 4 right). Two light sensors were added to the robot's front and back ends. Each light sensor was powered by a battery and connected to a Bluetooth audio jack, where a Bluetooth audio transmitter was connected.

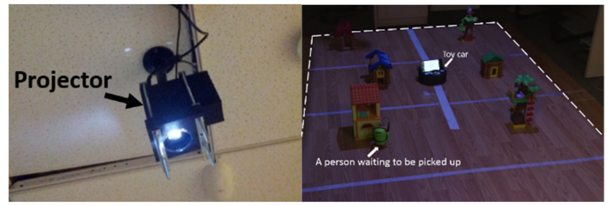

Figure 3 (Left) A projector is mounted on the room's ceiling. (Right) The floor set-up for the simulated toy car system.

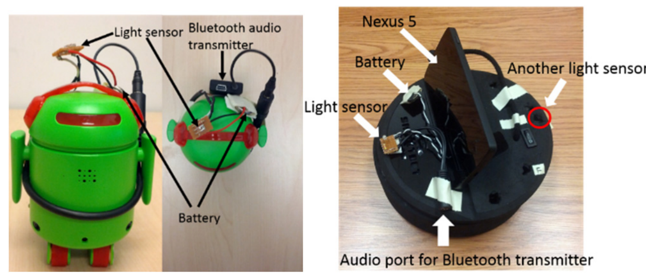

Figure 4 (Left) A customized robot simulates a passenger. (Right) Another customized robot simulates a toy car.

Figure 5 shows four snapshots of one trip of the toy car driving to pick up the "passenger". In this case, the passenger stands in front of a house and faces towards one road. The toy car first drives on the road labeled with wider white tape (the first and the second snapshots). At the intersection of two roads, the toy car stops and turns to its right 90 degrees (the second and the third snapshots). Finally, it drives along the new road and stops in front of "passenger" (The third and the last snapshots).

Figure 6 shows how the data flow and are processed in the system. The light sensor on the "passenger" reads the light intensity stream and sends the data stream using the Bluetooth audio transmitter a Bluetooth audio receiver on the server side. Similarly, two light sensors on the toy car read the light intensity streams from the projected light. The two Bluetooth transmitters connecting to the two light sensors send two steams of light sensor readings to another two Bluetooth audio receivers on the server. Based on these light sensor readings, The server computes the location of the "passenger" and also the location and the orientation of the toy car, and then sends these information back to the mobile phone installed on the along with trajectories of all roads. The mobile phone connects to and controls the toy robot car with an application. The application running on the phone plans the path to the "passenger" using the A* algorithm based on the received information and generates a sequence of motion commands which will be sent to and be executed by the robot car. The motion commands include moving forward and backward, rotating clock-wise or counter clockwise certain degrees.

The robot car might be off its moving direction slightly after driving a distance (*e.g.*, due to the slight difference of two motors that drive two wheels). Therefore, the car needs a real-time feedback to adjust its heading direction so as to keep

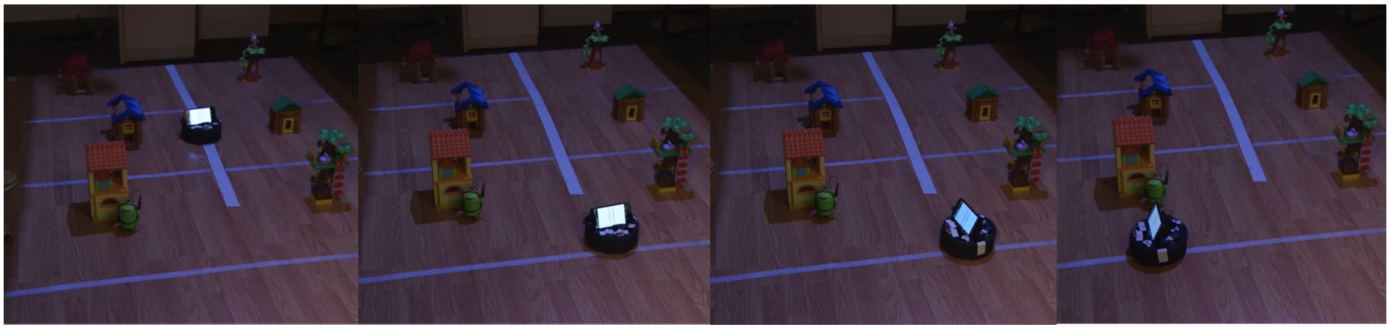

Figure 5 The toy car navigates on the road, turns to the left side and stops at the location of the passenger, which is simulated by a green *Bero* robot. On top of the robot's head, a light sensor is installed in order to receive the projected light from the overhead projector.

itself on "roads". The app running on the phone constantly computes its current heading direction and compares it with the current road's direction. If the offset exceeds a threshold, the app will issue a sequence of rotating command to the car to correct its heading gradually. For each rotating command, the toy car will only rotate slightly. By tuning the heading gradually will reduce the overshooting problem (rotating too much and then having to rotate back again). Because the localization technique provides its orientation and position more than 30 times per second, it allows the heading adjustment to happen in real time.

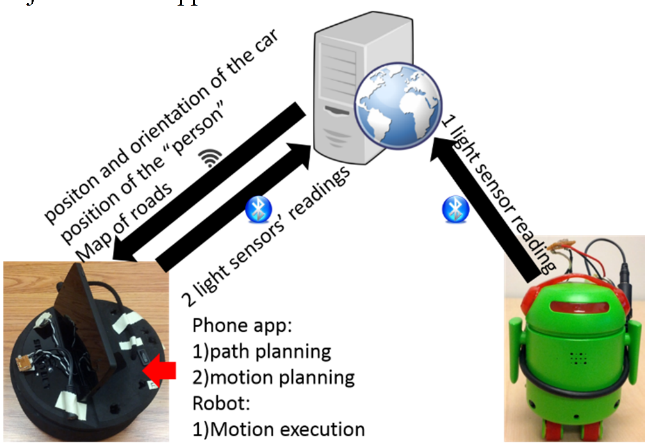

Figure 6 The data flow among different parts of the system.

IV. DISCUSSION AND FUTURE WORK

1) Multi-Object Localization Simultaneously. Because each toy car uploads its sensor readings to the server and get its location information back from server independently, multiple toy cars can compute their own routes locally and simultaneously. However, one challenge in a multi-car driving system is to take the "traffic" into consideration, which our current system does not address yet. By monitoring all the running cars' current locations and computed route for all toy cars, each car can then adjust its route accordingly to avoid the "collision" and the "traffic jam". **2) Extension of Localization Area**. Our system's current set-up supports the localization and tracking of multiple-object simultaneously in one projector's projection area, which is about a small office room size. A multiple-projector system can be designed to cover much bigger space. By assigning different projector with different sequences of gray images to project, we can make the coordinate system of each projector different from each other. One challenge is how to make the localization and tracking seamless across multiple projectors. One possible solution is to make two projectors' projection areas slightly overlapped. However, another issue would be how to deal with localization in the overlapping area where the light beams of multiple projectors are blended together. **3) Other applications**. The localization technique can be used to enable other applications. For example, multiple projectors might be installed on the ceiling of a shopping mall to replace the illumination sources. By utilizing the light sensor embedded on the front side of a phone, a shopping mall navigation app can be designed to enhance shopping experience.

V. CONCLUSION

In this paper, we introduce the design and implementation of a toy localization and navigation system. If embedded with two or more light sensors, the orientation of an object can also be determined. The projected light based indoor localization technique can localize multiple objects simultaneously in real time with fine granularity. Additionally, it does not require any pre-calibration and is perceived almost like a normal illumination source. By embedding tiny light sensors into everyday objects, the projected light localization technique can enhance the multiple smart objects interaction.

REFERENCES

[1] Bero.https://www.kickstarter.com/projects/realityrobotics/be-the-robot-bero-bluetooth-controlled-open-source

[2] G. Blanc, Y. Mezouar, , P. Martinet. 2005. Indoor navigation of a wheeled mobile robot along visual routes. In I*CRA*. 3354-3359.

[3] M. Fan, Q. Liu, H. Tang, P. Chiu. 2014. HiFi: hide and find digital content associated with physical objects via coded light. In *HotMobile'14. Article. 6.*

[4] B. Jones, H. Benko, E., Ofek, A. Wilson. 2013. IllumiRoom: Peripheral Projected Illusions for Interactive Experiences. In. *CHI'13*. 869-878.

[5] J. Lee, P. Dietz, D. Aminzade, S. Hudson. 2004 Automatic Projector Calibration using Embedded Light Sensors, In *UIST'04*. 123-126.

[6] D. Lymberopoulos, J. Liu, X. Yang, *et al*. 2015. A realistic evaluation and comparison of indoor location technologies: experiences and lessons learned. In. *IPSN'15*. 178-189.

[7] SmartBot. http://www.overdriverobotics.com/

[8] Y. Sugiura, D. Sakamoto, A. Withana, M. Inami, T. Igarashi. 2010. Cooking with robots: designing a household system working in open environments. In *CHI'10*. 2427-2430.

[9] Zhao, S., Nakamura, K., Ishii, K., Igarashi, T. 2009. Magic cards: a paper tag interface for implicit robot control. In *CHI*'09. 173-182.