# Indoor Toy Car Localization and Navigation using Projected Light

Authors Name/s per 1st Affiliation *(Author)*
line 1 (of *Affiliation*): dept. name of organization
line 2: name of organization, acronyms acceptable
line 3: City, Country
line 4: e-mail: name@xyz.com

Authors Name/s per 2nd Affiliation *(Author)*
line 1 (of *Affiliation*): dept. name of organization
line 2: name of organization, acronyms acceptable
line 3: City, Country
line 4: e-mail: name@xyz.com

*Abstract*— **Indoor localization is challenging in terms of both the accuracy and possible using scenarios. In this paper, we introduce the design and implementation of a toy car localization and navigation system, which demonstrates that a projected light based localization technique allows multiple devices to know and exchange their fine-grained location information in an indoor environment. The projected light consists of a sequence of gray code images which assigns each pixel in the projection area a unique gray code so as to distinguish their coordination. The light sensors installed on the toy car and the potential "passenger" receive the light stream from the projected light stream, based on which their locations are computed. The toy car then utilizes A\* algorithm to plan the route based on its own location, orientation, the target's location and the map of available "roads". The fast speed of localization 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 can power other applications that require fine-grained location information of multiple objects simultaneously.**

*Keywords-indoor localization; projected light; projector; multi-device interaction; navigation*

## I. INTRODUCTION

Location is one important context information that can benefit a wide range of applications. For instance, GPS can provide devices with their locations in the open outdoor environment. However, the accuracy of GPS is not enough for tasks such as self-driving car navigation. Additional equipment (*e.g.,* radar) is needed to provide more accurate location and local surrounding information. However, such techniques (*e.g*., the method used to drive the Google self-driving car) are not practical to adopt in an indoor environment. Additionally, the result of a recent indoor localization technique competition confirms that high accurate indoor localization is still challenging even today [6].

In this paper, we introduce the design and implementation of a toy car localization and navigation system, which demonstrates that a high precision indoor localization technique can enable multiple physical objects to be localized accurately and also to communicate with each other. An overhead projector 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. Furthermore, the toy car's position and orientation can be inferred as well. When a "person" 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 "person".

The projected light based localization technique and the client-server architecture can accomplish the following three challenging tasks simultaneously: 1) exchanging of high-precision location information among multiple devices (Toy car and the simulated "passenger") in real time; 2) real-time navigation guidance; 3) localization of multiple toy cars at the same time.

The toy car navigation application also demonstrates the following advantages of the projected light based localization technique. The projected light beam for localization is perceived as a constant-brightness illumination light source and does not change the environment appearance obviously. No calibration is required for localization. Because no camera is used, the privacy of any present people can be preserved.

By exploring the toy car localization and navigation, we also demonstrate that a potential usage of fine-granularity indoor localization technique is to enable intelligent interactions among multiple autonomous physical objects, It might shed light on the research of Smart Objects Interactions, where exchanging and understanding each other's information and knowledge are needed.

## 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
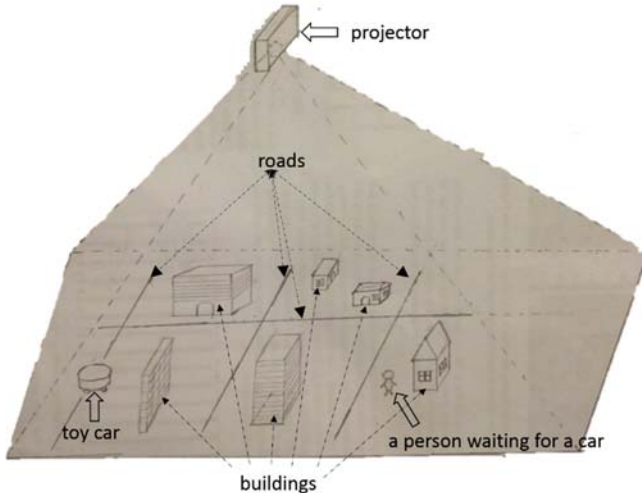


Figure 1. A sketch of the simulated environment for a toy car to pick up a "person". (Top) a projector projects the light beam to the floor and acts as an indoor GPS.



Figure 2. 6 gray code images (the first 3 encode the horizontal coordinate and the last 3 encode the vertical coordinate) which distinguish 64 positions (8 in horizontal * 8 in vertical).

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"). By exploring the toy car localization and navigation, we demonstrate that a potential usage of fine-granularity indoor localization technique is to enable intelligent interactions among multiple autonomous physical objects, which might inspire the research of Smart Objects Interaction and Internet of Things.

## 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

*1) Location:* The localization system uses a projector which periodically projects a sequence of binary images (Figure 2 shows an example sequence for a projection of 8 * 8 pixels). If these images are projected sequentially, each pixel of the projected area will receive a unique sequence of bright/dark light intensity values. If the bright and dark colors are coded as 1 and 0, then the binary sequence received at each pixel forms a gray code, which is unique in the projection area.

Let's use an example of a sequence of 6 binary images to understand the localization technique (Figure 2). When the 6 images are sequentially projected to a surface, the projected surface can be distinguished with 8 * 8 different coordinates. The first 3 images encode the horizontal coordinate, and the last 3 encodes the vertical coordinate. If we divide the horizontal and vertical edge of each image equally into 8 strips, then we will have 8 * 8 grids and each grid has a unique black and white sequence. For example, if 0 and 1 are used to represent black and white, then the 6 numbers used to represent the top left corner for two gray codes ((0, 0, 0), (0, 0, 0)), which is (0,0) in binary code. Similarly, the 6 numbers used for representing the bottom right corner for two gray codes ((1, 0, 0), (1, 0, 0)), which is (7, 7) in binary code. Generally speaking, for a projector with resolution of W * H, it requires to project $\lceil \log W \rceil + \lceil \log H \rceil$ gray code images sequentially to distinguish each pixel in the W*H projection area. By projecting such sequence of gray code images periodically in high frequency, each pixel in the projected area can be located many times per second which can be used for real-time applications. Therefore, any object in the projection area can be localized if equipped with a tiny light sensor.

*2) Orientation:* Our system can also compute the orientation of an object by using two geometrically separated light sensors. In Figure 3, two light sensors are installed on two ends of a toy car. 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 the 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 in hand 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 utile he/she reaches the other end. The app decodes all the positions
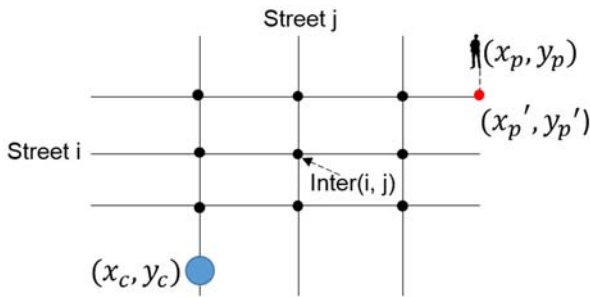
and stores them as the trajectory of that road. Then the person moves to another road and repeats the above procedure to construct the trajectory of another road. The procedure ends when the person finishes moving along all roads. Because our localization can decode about 80 positions per second, the 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 $(x_p, y_p)$ (see Figure 4). 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
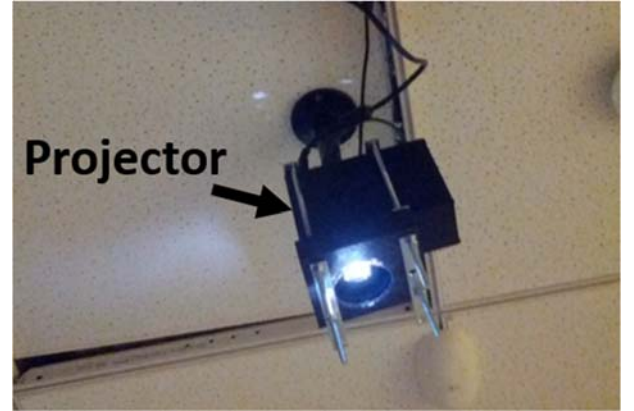


Figure 5. A projector is mounted on the room's ceiling.





Figure 4. Illustration of the navigation map.

Figure 6. The floor set-up for the simulated toy car system.
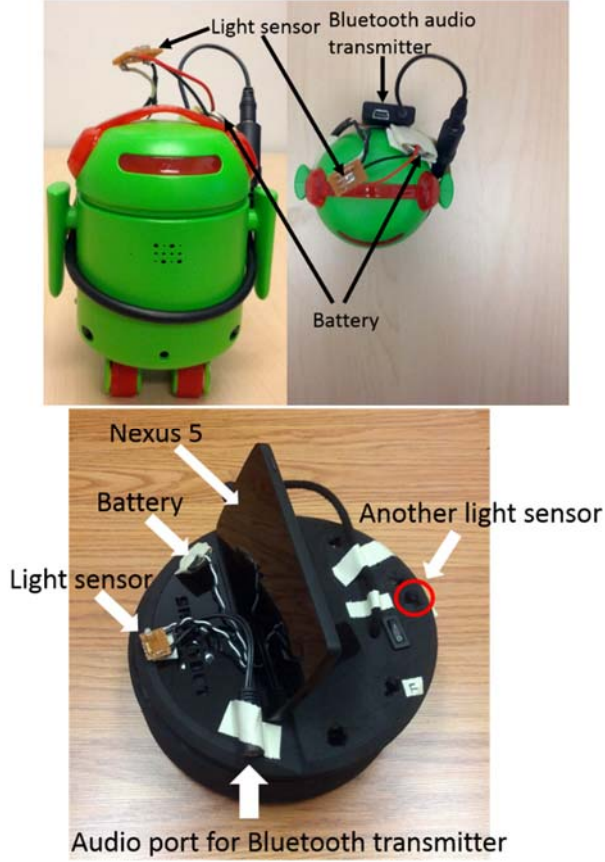Figure 3. Two light sensors are installed on two sides of a robot.

Figure 7 (Top) A customized robot simulated a person; (Bottom) A customized robot simulated a toy car.

and choose the next intersection for the planned route until we reach the last intersection that is closest to the "person.

### E. Implementation Details

We fully implemented the toy car localization and navigation system a deployed in a small office environment. A projector was mounted on the ceiling of the office (Figure 5) and the floor set up was shown in Figure 6.

The white dash lines in Figure 6 highlighted the three of four boundaries of the projection area. As noticed 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 7 Top). 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
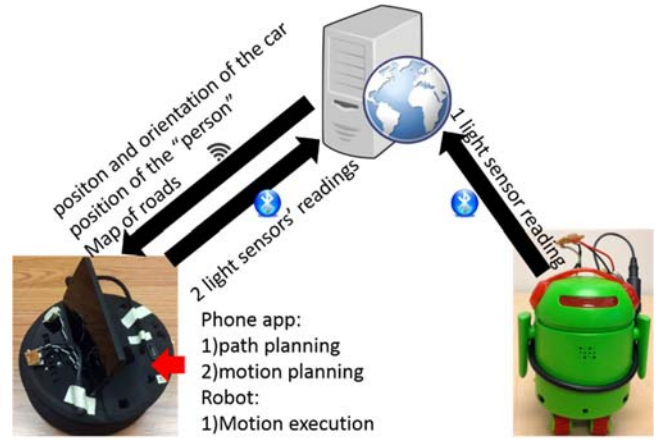


Figure 9 How data flow and processed in the system.

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 7 Bottom). 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 8 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 in Figure 8). At the intersection of two roads, the toy car stops and turns to its right 90 degrees (the second and the third snapshots in Figure 8). Finally, it drives along the new road and stops in front of "passenger" (The third and the last snapshots in Figure 8).

Figure 10 The toy car compares its heading direction with the road direction, and adjusts its heading direction by rotating gradually to align with the road when it exceeds a threshold.

Figure 9 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 itself on "roads". Figure 10 shows two snapshots of the toy car

tries to adjust its heading when it computes that the angle between its heading orientation and the road exceeds a threshold. 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.

## IV. DISCUSSION AND FUTURE WORK

In this section, we will discuss the potential problems and the solutions and present the future work:

### A. 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



Figure 8 The toy car navigates on the road, turn 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 installed in order to receive the projected light from the overhead projector.

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".

### B. 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 the localization in the overlapping area where the light beams of multiple projectors might be blended together.

### C. Other Potential Applications

Toy car navigation demonstrates the fast speed and fine granularity of projected light based localization technique. The localization technique can be used to enable other applications which require fine-grained location information of multiple devices or objects. For instance, multiple projectors might be installed on the ceiling of a shopping mall to replace the illumination sources (*e.g.*, light bulbs). By leveraging the light sensor embedded on the front side of mobile phones, a fast and high accurate indoor localization can be designed for shopping navigation scenario.

## 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 *Proceedings of IEEE International Conference on Robotics and Automation. (ICRA)*. 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 *Proceedings of International Workshop on Mobile Computing Systems and Applications.. HotMobile'14. Article. 6*.

[4] B. Jones, H. Benko, E., Ofek, A. Wilson. 2013. IllumiRoom: Peripheral Projected Illusions for Interactive Experiences. In *Proceedings of ACM Conference on Human Factors in Computing Systems.. CHI'13*. 869-878.

[5] J. Lee, P. Dietz, D. Aminzade, S. Hudson. 2004 Automatic Projector Calibration using Embedded Light Sensors, In *Proceedings of ACM Symposium of User Interface Software and Technology.. 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 *Proceedings of International Conference on Information Processing in Sensor Networks. 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 *Proceedings of ACM Conference on Human Factors in Computing Systems. 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 *Proceedings of ACM Conference on Human Factors in Computing Systems. CHI'09*. 173-182.