

# POLI: INTERACTIVE MULTIMEDIA AUTHORIZING AND RETRIEVAL ON MOBILE PHONE BY LEVERAGING ITS AUDIO CHANNEL AND CODED LIGHT

Anonymous submission

## ABSTRACT

Connecting digital information with physical objects can enrich their content and make them interactive. In this paper, we present POLI, which allows a user to author digital content to physical objects using a mobile authoring app and another user to explore the authored content using a mobile client app. The mobile apps discover their user's location by decoding a gray code sequence received from a light sensor which is plugged in the phone's audio jack. To make authoring process natural and easy, POLI provides three novel gesture based interactive approaches, which follow people's intuitive understandings of an object's boundary.

**Index Terms**— Mobile authoring tool, gesture authoring, coded light, interactive environment.

## 1. INTRODUCTION

Augmenting physical objects with digital content may enrich their user experience. Traditional augmented reality (AR) approaches (*e.g.*, NaviCam [10], CyberCode[11]) leverage visual markers to define locations for the augmented content. However, visual markers may change the natural appearance of physical environment or objects. There are also approaches using image processing to define the regions for augmented content in the live camera view without visual markers (*e.g.*, Handy AR [8]). However, the robustness and accuracy of image processing approaches may be affected by environment light condition or clutter backgrounds.

Without using markers or image processing approaches, Fan *et al.* [1] recently proposed a novel system, called "HiFi", which used a projector and a camera together for defining interactive regions for physical objects (hotspot). The authoring process was conducted on a PC, where the camera and the projector were installed. The camera was set to view the entire projection area so that the transformation matrix between the camera's and the projector's views can be inferred. A system administrator defined the region of a hotspot by dragging a bounding box around an object using a mouse in the camera's view. Then the coordinate of the bounding box was translated into the projector's coordinate system and used as the effective area of the hotspot. There are three limitations about the authoring approach of HiFi. First, configuring the camera and the project's views so that the

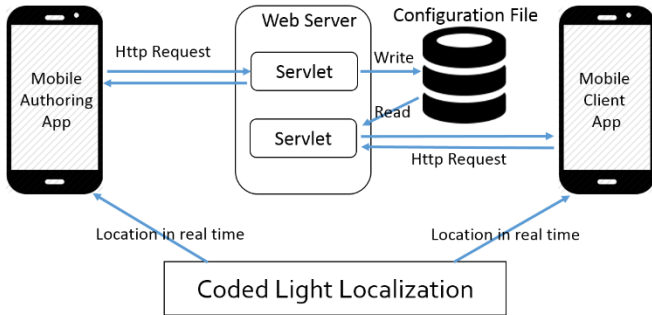
prior one covers the latter can be time-consuming or sometimes even impossible. Second, dragging bounding boxes around objects in the camera's view using a mouse may be unnatural and inaccurate because of the camera's resolution, the viewing angle and the shape of the object. Lastly, the physical authoring region is restricted by the camera's view. The objects in the projection area but not in the camera's view cannot be authored.

In this paper, we present POLI, an interactive authoring and retrieval system on mobile phones. POLI extends "HiFi" and solves the aforementioned three limitations. POLI leverages a mobile phone that has a tiny light sensor plugged in its audio jack for authoring and retrieving. Such design enables the mobile phone to localize itself in real time on its own. It does not require a central camera, and authoring and retrieving can happen in front of real objects in a 3D space. In addition, POLI provides three natural interactive approaches for users to easily define the boundaries of physical objects.

## 2. RELATED WORK

Visual markers have been used extensively in AR to define locations where the augmented content should be shown, for example, NaviCam [12], CyberCode [11] and Tiles [10] (more extensive reviews can be found in Azuma's [1], Azuma *et al.*'s [2], Krevelen and Poelman's [5] survey papers). Even though less obtrusive visual markers were studied [16], markers in general still changed the appearance of physical objects or environment. In contrast, POLI used a projector to project a beam of specially design light which is perceived as a normal illumination light.

Objects recognition on camera's video frames can be used to replace visual markers for defining authoring locations. For example, "Handy AR" [8] recognized and tracked fingertips where augmented content was shown. However, the accuracy and robustness of image recognition depend on not only the algorithm design but also physical conditions (*e.g.*, light condition, clutter background, orientation and occlusion of objects). On the other hand, POLI leverages the gray code sequence to define a target object area which is robust to the environmental conditions (*e.g.*, the light and the object's orientation) and also much faster than normal image processing approaches.



**Fig. 1.** The architecture of POLI interactive multimedia authoring and retrieval system.

Sensor based approaches may also help with the authoring. For example, iCam [9] used a sensor-rich handheld device and an indoor localization system [15] to localize itself and also its surrounding space so as to know the target authoring location indicated by a beam of laser light from the user’s handheld device. In contrast, POLI has coded light projected through the space as the only infrastructure for localization.

On the other hand, there are few studies looking at making the authoring process natural and fun. Lee *et al.* [6] presented a tangible method for authoring digital content by manipulating a physical cube. POLI enriches the user experience of authoring process by presenting three interactive approaches for defining the effective area of a hotspot, which utilizes natural hand gestures and follows people’s natural understanding of an object’s boundary.

### 3. POLI DESIGN AND IMPLEMENTATION

POLI is an interactive multimedia authoring and retrieval system using mobile devices. The architecture of POLI is shown in Fig. 1. It has four basic components. The Coded Light Localization unit consists of a projector which constantly projects coded light to the target authoring region and acts as the infrastructure for localization. A mobile authoring app allows a system administrator to define hotspots and author digital content on them. A mobile client app allows end users to retrieve the digital content from hotspots. Both the mobile authoring app and the mobile client app have a thread which reads the light intensity value sensed by the external light sensor plugged in the phone’s audio jack and decodes the phone’s current position. The Web Server runs on a series of servlets. The Web Server receives configuration file that describes all defined hotspots from a mobile authoring app. It also response to a mobile client app’s HttpRequests to provide the configuration file, which will help the mobile client app reason whether and what digital content to show. In the following subsections, we will explain in details about each component of POLI.

#### 3.1. Real-time Localization on Mobile Phone using Coded Light and Phone’s Audio Channel



**Fig. 2** Six Gray code images (the first three to distinguish the horizontal coordinate and the last three for the vertical coordinate) which can distinguish 64 positions (8 in horizontal \* 8 in vertical).

The localization system uses a DLP projector which periodically projects “Coded Light”. Coded light contains a sequence of gray code images. If these images are projected sequentially, each pixel of the projected area will receive a different and unique sequence of bright/dark light intensity values. If bright and dark are coded as 1 and 0, then the sequence forms a gray code. Lee *et al.* [7] leveraged a similar technology to calibrate the projection onto any arbitrary size and oriented surface. Jones *et al.* [4] used gray code images to figure out the layout around a TV in order to project game content beyond the TV’s screen. Schmidt *et al.* [13] used gray code images to infer a light sensor’s coordinate inside a projection so as to active different GUI elements.

Fig. 2 shows a sequence of 6 gray code images. When they are sequentially projected to a surface periodically, the projected surface can be distinguished with  $8 * 8$  different coordinates. The first three images are used to encode the horizontal coordinate, while the last three are used for encoding the vertical coordinate. If we divide each image equally into  $8 * 8$  grids, then each grid has a unique coordinate. For instance, if we use black for 0 and white for 1, then the gray code for the top left corner is  $((0, 0, 0), (0, 0, 0))$ , which is  $(0, 0)$ . In the same way, the gray code of the bottom right corner is  $((1, 0, 0), (1, 0, 0))$ , which is  $(7, 7)$ . Generally speaking, it requires to project  $[logW] + [logH]$  gray code images sequentially to distinguish each pixel in a  $W * H$  projection area. By projecting such a sequence of gray code images periodically, each pixel in the projected area can be located many times per second which can be used for real-time applications.

A light sensor can sense the light intensity change. However, to capture the fast change in the readings of a light sensor, we need to sample the readings of a light sensor in high speed. We finally leverage the audio channel of a mobile phone and come up with the design shown in Fig. 3. Using the phone’s audio channel for sensing other types of data has been explored in commercial products as well, such as square [14]. A light sensor unit is attached to the top front part of a



**Fig. 3** To localize a phone, a light sensor is plugged into the audio jack of an android phone (Galaxy Nexus) for sampling the coded light in real time, which are then feed to a decoding algorithm.

phone and plugged into its audio jack. When such combined phone and light sensor unit is placed into the “Coded Light” projection area, the location decoding algorithm running in the mobile authoring app and the client app will sample the readings of the light sensor from the audio channel as an analog signal stream with high sampling rate (e.g. 44K Hz) and digitize them. The decoding algorithm finds the starting and ending bits, and then extracts the embedded gray codes in between. For the area projected by a projector with the resolution of  $W * H$ , there are  $\lceil \log W \rceil + \lceil \log H \rceil$  embedded gray codes which represent the location  $(x,y)$ ,  $0 \leq x < W, 0 \leq y < H$ .

### 3.2 Hotspot in Physical Space

We define a hotspot as a region or an object in physical space where digital content have been assigned. A user can interact with digital content when she enters the physical boundary of a hotspot using her mobile phone. For instance, a hotspot can be an art painting hanging on a wall where a video is assigned within its physical boundary. When a user’s mobile phone enters the painting’s physical region, the authored video will be retrieved and played on her mobile phone. In addition, a hotspot may also be a region where a device control is assigned. For instance, in a home environment, when a user moves her phone into the physical region of a lamp whose light intensity control is authored, her phone may retrieve the lamp light intensity control and shows her an interactive GUI for tuning the light intensity on her phone.

In summary, a hotspot in physical space is an object or a region where a digital content or an interaction is authored so as to enhance its user experience. A user can interact with a hotspot once her mobile device enters the physical boundary of the hotspot.

### 3.3. Defining a Hotspot

To facilitate the process of defining hotspot, POLI allows a system administrator to define a hotspot simply using her mobile phone with hand gestures. One property of characterizing a hotspot (e.g., a physical object or a region) is its effective area in the physical space. Therefore, defining a

hotspot in a physical object or a region is a procedure of associating digital content with a physical area.

#### 3.3.1. Interactive Ways of Defining a Hotspot Region

As described in section 3.1, the location decoding algorithm on mobile phone calculates the phone’s location in real time by decoding a sequence of light intensity values of Coded Light received from a light sensor that is plugged in the phone’s audio jack. To further define the region of a hotspot, we need an interactive approach to define a region where digital information or interaction could happen using point locations decoded by the decoding algorithm. We have explored the following three possible authoring approaches (Figure 3) for the mobile authoring app.

- *Center + Radius Authoring*

“Center + Radius Authoring” defines the region of a hotspot by its center and an effective radius R (Fig. 4 (a)). The actual POLI Mobile authoring UI for this method is shown in Fig. 5 (a). When a user moves a phone to the center of a hotspot and clicks the “center” button on the GUI, the position decoded at that moment will be stored as the center’s coordinate. When the user then moves the phone to the edge of the imaginary circle for the hotspot and clicks the “edge” button, the position of the edge point will be decoded and then the distance between edge point and the center point is calculated and used as the effective radius R of the hotspot. When a client user’s mobile device is inside the circle area of the painting, the authored content (e.g., a video) will be retrieved and played by the POLI mobile client app in real time.

- *Scribble Authoring*

“Scribble Authoring” is illustrated in Fig. 4 (b), which defines the region of a hotspot by a sequence of points on the trace scribbled by a user. The actual POLI mobile authoring UI is shown in Fig. 5 (b). With this app, a user can define the area of a hotspot by pressing down the “Scribble” button, and scribbling over the hotspot area before releasing the button. The app then records a sequence of decoded positions that represent the scribbled trace. We define the effective area of

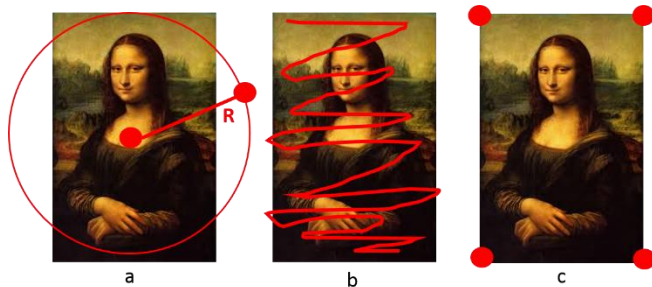


Fig. 4. Illustration of three authoring mechanism with a painting: (a) Center + Radius authoring; (b) Scribble authoring; (c) Four-Corner authoring



Fig. 5. Three mobile authoring UIs: (a) Center + Radius authoring; (b) Scribble authoring; (c) Four-Corner Authoring

a hotspot as the minimum bounding box of the scribbled trace. Let's assume the coordinate of the sequence of positions from time  $i$  to time  $j$  are represented as  $(t, x_t, y_t)$ ,  $t = i, \dots, j$ . the minimum bounding box of the scribbled trace is computed and represented by four lines, two horizontal and two vertical lines:  $x_{min} = \min x_t$ ,  $x_{max} = \max x_t$ ,  $y_{min} = \min y_t$ ,  $y_{max} = \max y_t$ ,  $t = i, \dots, j$ . If a client user's mobile phone's position  $(x, y)$  satisfies the constrains:  $x_{min} \leq x \leq x_{max}$  &  $y_{min} \leq y \leq y_{max}$ , the authored digital content will be retrieved and played on the phone.

It is important to point out that one special case of scribble authoring is moving a phone around the target object by following its boundary lines. The authored region will reflect exactly the object's actual shape.

- *Four-Corner Authoring*

The idea of "Four-Corner Authoring" is illustrated in Fig. 4 (c), which defines the area of a hotspot by the four corners of a rectangle. The POLI mobile authoring app has four buttons to let a user press and define the coordinate of the four corners (Fig.5 (c)). If a client user's mobile phone is in the rectangle area defined by the four corners, then the digital content will be retrieved and played on the phone. Let's define the coordinate of the top-left, the top-right, the bottom-left and the bottom-right corners as the following  $(x_{tl}, y_{tl})$ ,  $(x_{tr}, y_{tr})$ ,  $(x_{bl}, y_{bl})$ ,  $(x_{br}, y_{br})$  and the coordinate of the client user's mobile phone as  $(x, y)$ . If  $(x, y)$  satisfies the following two constrains:

$$\begin{aligned} & ((y_{tr} - y_{tl}) * (x - x_{tl}) - (x_{tr} - x_{tl}) * (y - y_{tl})) * \\ & ((y_{br} - y_{bl}) * (x - x_{bl}) - (x_{br} - x_{bl}) * (y - y_{bl})) \leq 0 \\ \text{and} & ((y_{bl} - y_{tl}) * (x - x_{tl}) - (x_{bl} - x_{tl}) * (y - y_{bl})) * \\ & ((y_{br} - y_{tr}) * (x - x_{br}) - (x_{br} - x_{tr}) * (y - y_{br})) \leq 0 \end{aligned}$$

, the phone is inside the hotspot and the authored digital content will be retrieved and played on the phone.

### 3.3.2.. Creating & Uploading the Hotspot Configurations

After defining the effective area of a hotspot, the next step is to choose the type of digital content to author to the hotspot.

```
{
  "tl_x": *, "tl_y": *, "tr_x": *, "tr_y": *, "bl_x": *, "bl_y": *, "br_x": *, "br_y": *,
  "type": *, "mmname": *}

```

**Fig. 6** The configuration of a hotspot in JSON format: The first eight parameters are used to define the coordinates of four corners. The "type" defines the type of the digital content, which can be one of the following four: "VIDEO","IMAGE" "AUDIO" and "TEXT". The "mmname" contains the file name of the digital content.

The mobile authoring app enables a system administrator to choose a multimedia file (e.g., video, audio or a text) from the authoring phone. Once the digital content is selected, a configuration for the hotspot is created as a JSON string. The format of the JSON string created with the "Four-Corner Authoring" method is defined in Fig. 6. The generated JSON string and the actual file indicated in the "mmname" field of the JSON string from the configuration for a hotspot. POLI allows a user to define as many hotspots as possible. The JSON string for each hotspot will be combined together as a JSON array. This JSON array and all the multimedia content files described in each JSON string's "mmname" will be uploaded to POLI web server via HttpRequests.

### 3.4. Interacting with a Hotspot

The goal for authoring hotspots is to let visitors explore and interact with them. For explorations and interactions, a client user needs a mobile phone plugged with a light sensor (Fig. 3) and the mobile client app. When the phone enters a hotspots area, the mobile client app will send a HTTP request to the Web Server to fetch the configuration file, which describes all the existing hotspots (e.g., their locations, effective areas and the authored digital content). The mobile app constantly decodes the phone's current location based on the light intensity changes sampled from the phone's audio jack, and computes whether it is inside one of the hotspots. Once the mobile phone is in a hotspot, the Mobile App will send another HttpRequest to the Web Server to fetch the corresponding digital content. For instance, if the digital content is a video, then the video will be streamed from the server to the mobile phone, where the user can view it.



**Fig. 7.** Simulated mini Museum Environment Set-up in a room: (a) A projector is mounted on the top side of a wall and is projecting the coded light to the opposite side wall. (b) Two paintings are hang in the projection area and the museum registration paper is placed on a table.

#### 4. APPLICATION

In this section, we introduce a concrete application of POLI, a *Museum Assistant* system. To demonstrate the *Museum Assistant* concept, we simulated a mini-museum in a room (Fig. 7). The mini Museum included a projector which was mounted on the top end of a vertical wall and projected the coded light to the opposite side vertical wall with the two paintings.

When a system admin enters the projection area and launches the *MuseumAuthoring* app on an authoring phone that has a light sensor plugged in the audio jack, she will see the interface shown in Fig. 8 (a). In current implementation, she needs to input an IP address to identify the Web Server, but it could be an easy-to-remember name (e.g., the name of the room) and the system will figure out the corresponding web server. Then she can choose her favorite way to define the hotspot regions from the three options that we have described in Section 3.3 using interface Fig. 8 (b). Then she can choose a multimedia content from her phone by pressing “Click to choose a video to author” button on Fig. 8 (c) and she will have a chance to preview the content. For instance, if it is a video, then she gets a chance to preview the video content in the “Video View”. If she is satisfied with the video, she can press “Next” button to proceed to the next screen (Fig. 8 (d)). Meanwhile, the video file will be uploaded to the Web Server. After that, she can continue to author another hotspot by pressing the “Continue to author another hotspot” button, which will take her back to the interface Fig.8 (b). In the meantime, the configuration of this hotspot is created as a JSON string as described in Section 2.2.3 and the JSON string will be added into the JSON array. In other words, the JSON array contains the configuration for all hotspots that have been defined so far. If she has finished all the authoring, she can simple press the “Finish all the authoring” button, when the JSON array will be uploaded to the Web Server as the configuration for all the authored hotspots.

On the other hand, a visitor can use a mobile phone to retrieve and view the digital content from hotspots. To prevent unauthorized access to our system, a visitor needs to first register a phone to the system. To do so, the visitor just needs to start a client app and places the phone in the

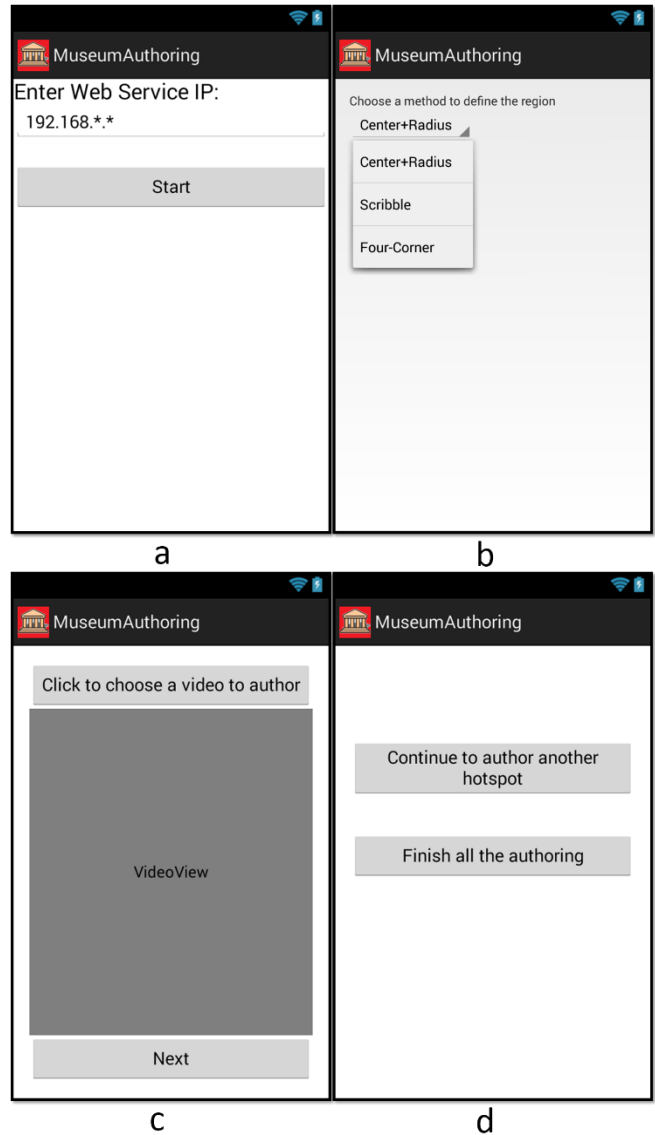


Fig. 8. Mobile Authoring Steps for the Museum Assistant System.

registration paper area, which is placed on the surface of a table (Fig. 7 (b) and Fig. 9 (a)). The client app will send the phone’s location to the web Server for verification. Once the Web Server confirms that the phone is placed in the

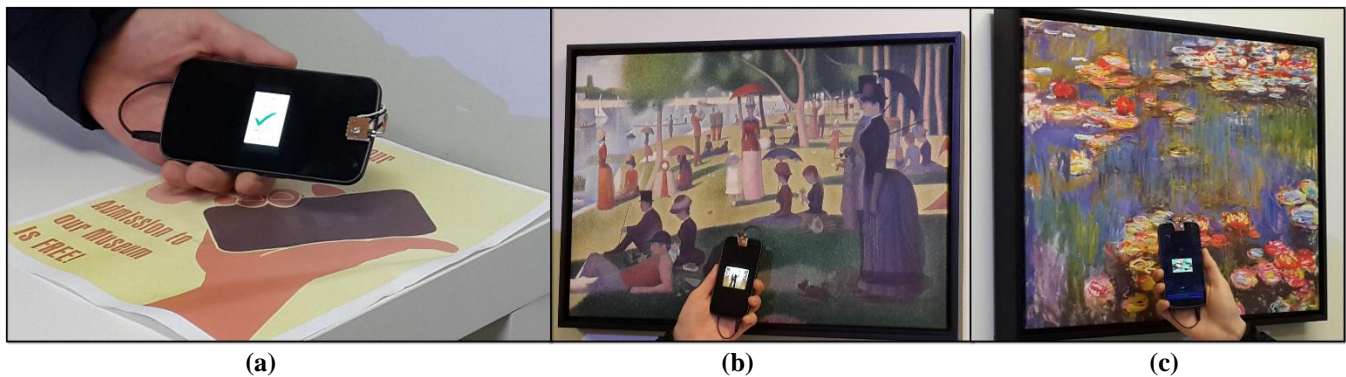


Fig. 9. A visitor registers to the mini-museum (a), and discovers the authored digital content for two paintings using a mobile phone (b, c).

registration paper area, it will grant this mobile phone access to the *Museum Assistant* system and also send the configuration of all hotspots to the client app. Once the client app gets the configuration file, it can be aware of all the hotspots and the authored digital content. The client app constantly computes the phone's position and checks if the position is inside one of the hotspot area while the visitor walks around in the museum. If yes, then the client app will send the Web Server another `HttpRequest` to fetch the corresponding digital content. In this case, a video is streamed to the phone and shown on the screen (Fig. 9, b and c). Note that the videos are different in Fig. 9 b and c, because the visitor's phone is in the hotspot area of different paintings.

## 5. DISCUSSION

### 5.1. Multi-user Support

POLI allows multiple users to interact with hotspots simultaneously, because the mobile client app running on each user's phone computes its location independently and also interacts with the web server independently. However, POLI does require a line-of-sight between the light sensor and the projector's light beam.

### 5.2. Roles of Authoring and Retrieving

Although the roles of authoring and retrieving are assigned to two different groups of users (the system admin and the visitors) in the *Museum Assistant* application scenario, POLI can also allow a user to both author and retrieve at the same time using the same mobile app.

### 5.3. Access Control to Hotspots

Current system allows all the registered users to discover all authored hotspots. However, we can assign different access rights if necessary (*e.g.*, by adding a field in the configuration file to label user's identity). Imagine an office with POLI set-up and projecting light beam to staffs' desks, Amy can define her desk area as her private zone where only she can retrieve any authored content (*e.g.*, invisible digital notes authored by others using POLI mobile authoring app).

## 6. CONCLUSION AND FUTURE WORK

In this paper, we present POLI, which allows a user to author hotspots to physical objects and client users to explore the hotspots using mobile phones. POLI mobile apps detect a user's location in real time by sampling a light sensor's readings from the phone's audio jack. POLI provides three gesture based interactive ways for authoring which follow intuitive understandings of an object's boundary and easy to perform. The coded light used in POLI is perceived by users as another illumination source. Therefore, POLI does not change the natural appearance of them. In addition, POLI supports multi-user interacting with hotspots simultaneously. POLI also allow a user to author and retrieve content at the same time.

In our current implementation, all interactions happen in one projector's projection area. For a large room, it would be challenging to cover the entire space. Therefore, it would be interesting to explore how to extend the interaction to a much broader space (*e.g.*, a large room). Multiple projectors could be used, where each would be assigned a non-overlapped gray code images sequences for projection.

## 7. REFERENCES

- [1] R. T. Azuma. (1997). A survey of augmented reality. *Presence*, 6(4), 355-385.
- [2] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, B. MacIntyre. (2001). Recent advances in augmented reality. *IEEE Transaction on Computer Graphics and Applications*, 21(6), 34-47.
- [3] M. Fan, Q. Liu, H. Tang, P. Chiu. (2014). HiFi: hide and find digital content associated with physical objects via coded light. In *Proc. HotMobile'14*. Article 6.
- [4] B. Jones, H. Benko, E. Ofek, A. Wilson. (2013) IllumiRoom: Peripheral Projected Illusions for Interactive Experiences. In *Proc. CHI'13*. 869-878.
- [5] D.W.F. Krevelen, R. Poelman. A Survey of Augmented Reality Technologies, Applications and Limitations. *International Journal of Virtual Reality*, 2010, 9(2):1-20.
- [6] G. A. Lee, C. Nelles, M. Billinghurst, G. J. Kim. (2004). Immersive authoring of tangible augmented reality applications. In *Proc. ISMAR'04*. 172-181.
- [7] J. Lee, P. Dietz, D. Aminzade, S. Hudson. (2004) Automatic Projector Calibration using Embedded Light Sensors, In *Proc. UIST'04*. 123-126.
- [8] T. Lee, T. Hollerer. (2007). Handy AR: Markerless inspection of augmented reality objects using fingertip tracking. In *Proc. ISWC'07*. 83-90.
- [9] S. Patel, J. Rekimoto, G. Abowd, (2006). icam: Precise at-a-distance interaction in the physical environment. In *Proc. Pervasive Computing'06*. 272-287.
- [10] I. Poupyrev, D. Tan, M. Billinghurst, H. Kato, H. Regenbrecht, N. Tetsutani. (2001). Tiles: A mixed reality authoring interface. In *Proc. INTERACT'01*. 334-341.
- [11] J. Rekimoto, Y. Ayatsuka. (2000). CyberCode: designing augmented reality environments with visual tags. In *Proc. of DARE 2000 on Designing augmented reality environments*. 1-10.
- [12] J. Rekimoto, K. Nagao. (1995). The world through the computer: Computer augmented interaction with real world environments. In *Proc. UIST'95*, 29-36
- [13] D. Schmidt, D. Molyneaux, X. Cao. (2012). PIControl: using a handheld projector for direct control of physical devices through visible light. In *Proc. UIST'12*. 379-388.
- [14] Squareup. <https://squareup.com/reader>
- [15] UbiSense. <http://ubisense.net/en>
- [16] D. Wagner, T. Langlotz, D. Schmalstieg. (2008). Robust and unobtrusive marker tracking on mobile phones. In *Proc. ISMAR'08*. 121-124.