

Multilingual Digital Signage using Computer Vision and Bluetooth Beacons

A Thesis Presented to
The Faculty of the Computer Science Department
California State University Channel Islands

In (Partial) Fulfillment
of the Requirements for the Degree
Masters of Science in Computer Science

by

Suhas Dwarakanath

Advisor: Dr. Brian Thoms and Dr. David Claveu

October 2018

© 2018
Suhas Dwarakanath
ALL RIGHTS RESERVED

APPROVED FOR MS IN COMPUTER SCIENCE

Advisor: Dr. Brian Thoms and Dr. David Claveu Date

Name _____ **Date** _____

Name _____ **Date** _____

APPROVED FOR THE UNIVERSITY

Name _____ **Date** _____

Non-Exclusive Distribution License

In order for California State University Channel Islands (CSUCI) to reproduce, translate and distribute your submission worldwide through the CSUCI Institutional Repository, your agreement to the following terms is necessary. The author(s) retain any copyright currently on the item as well as the ability to submit the item to publishers or other repositories.

By signing and submitting this license, you (the author(s) or copyright owner) grants to CSUCI the nonexclusive right to reproduce, translate (as defined below), and/or distribute your submission (including the abstract) worldwide in print and electronic format and in any medium, including but not limited to audio or video.

You agree that CSUCI may, without changing the content, translate the submission to any medium or format for the purpose of preservation.

You also agree that CSUCI may keep more than one copy of this submission for purposes of security, backup and preservation.

You represent that the submission is your original work, and that you have the right to grant the rights contained in this license. You also represent that your submission does not, to the best of your knowledge, infringe upon anyone's copyright. You also represent and warrant that the submission contains no libelous or other unlawful matter and makes no improper invasion of the privacy of any other person.

If the submission contains material for which you do not hold copyright, you represent that you have obtained the unrestricted permission of the copyright owner to grant CSUCI the rights required by this license, and that such third party owned material is clearly identified and acknowledged within the text or content of the submission. You take full responsibility to obtain permission to use any material that is not your own. This permission must be granted to you before you sign this form.

IF THE SUBMISSION IS BASED UPON WORK THAT HAS BEEN SPONSORED OR SUPPORTED BY AN AGENCY OR ORGANIZATION OTHER THAN CSUCI, YOU REPRESENT THAT YOU HAVE FULFILLED ANY RIGHT OF REVIEW OR OTHER OBLIGATIONS REQUIRED BY SUCH CONTRACT OR AGREEMENT.

The CSUCI Institutional Repository will clearly identify your name(s) as the author(s) or owner(s) of the submission, and will not make any alteration, other than as allowed by this license, to your submission.

Title of Item

3 to 5 keywords or phrases to describe the item

Author(s) Name (Print)

Author(s) Signature

Date

Multilingual Digital Signage using Computer Vision and Bluetooth Beacons

Suhas Dwarakanath

October 3, 2018

Abstract

Due to globalization and changing lifestyle, more and more people are visiting foreign countries for business and travel. Also lately, a lot of newly arriving refugee families to the U.S face legal consequences. One of the struggles they face is reading documentation they receive through mail; whether bills, court documents or financial assistance documents, they struggle to read and understand them. There are thousands of languages in the world and it is impractical to install signage and print documents in all the languages. In this research, by combining Computer Vision and Bluetooth beacons, multilingual digital information is displayed on the user's smartphone. Smartphone camera allows the user to take a picture of a document. It is then posted to the Google Cloud Vision API which returns the text of that document. It can then be translated to any language using Google Translate API. The system also displays the information of nearby signages (with bluetooth beacons) on the smartphone. This system was implemented in the university campus and the evaluation experiment was conducted by on international students. It was found that the system helps the users to understand their environment better in their native language.

Contents

1	Introduction	1
2	Background	2
2.1	Related Work	2
2.2	Outline	5
2.2.1	GPS	6
2.2.2	Wi-Fi	6
2.2.3	Near Field Communication (NFC)	6
2.2.4	Bluetooth	6
2.2.5	iBeacon	7
2.2.6	Comparison of indoor positioning methods	7
3	Understanding iBeacon Technology	9
3.1	Existing applications implemented using iBeacon technology .	10
3.2	The algorithm to calculate distance based on RSSI	11
4	Google Vision API	13
4.1	Sample Request	15
4.2	Sample Response	16
5	Google Cloud Translation API	17
5.1	Sample Request	18
5.2	Sample Response	19
6	Algorithm	20
7	System Design	22
7.1	Environment	23
7.1.1	Digital Signages	23
7.1.2	Printed Signages	24
7.2	Mobile application	24
7.2.1	Background Monitoring	25
7.2.2	Beacon Ranging	25
7.2.3	Development stack	25
7.2.4	Screenshots	27
7.3	REST API Integration	33

7.4	Database	33
8	Designing the User Experience	36
8.1	User Analysis	37
8.1.1	Example Persona	37
8.2	Task Analysis	38
8.3	Context Analysis	39
8.4	Design Principles	39
8.4.1	Affordance	39
8.4.2	Visibility	40
8.4.3	Constraint	41
8.4.4	Consistency	41
8.4.5	Feedback	41
9	Case study	43
9.1	Digital Signages	43
9.1.1	Test case 1	43
9.1.2	Test case 2	44
9.2	Printed Signages	45
9.2.1	Test case 3	46
9.2.2	Test case 4	48
9.3	Testing environment	51
10	Conclusion and future work	53
	References	56

List of Figures

1	iBeacon packet format	9
2	UML activity diagram for Text Detection using Google Vision API	14
3	Example image for testing Google Vision API	15
4	UML activity diagram using Google Translation API	18
5	Process Flow Diagram	23
6	Home page	27
7	Text extraction	27
8	Translation	28
9	Share text feature	28
10	Type and translate text feature	30
11	Digital Signage	31
12	Translate feature	31
13	Text translated to Spanish language	32
14	Database Schema	34
15	Affordance design principle	40
16	Feedback	42
17	Digital signage discovered at 1.8 meters away	45
18	Printed Signage 1	47
19	OCR feature	47
20	Text translated to English	48
21	Printed Signage 1	50
22	OCR feature	50
23	Text translated to English	51

1 Introduction

With changing lifestyle and globalization, more people visit foreign countries. Every country is working towards becoming tourism oriented to increase their economy. Most of the visitors use their smartphone to access information. However, when visiting foreign countries, most people face difficulties in obtaining information due to difference in language, which makes it an inconvenience to stay in foreign countries for a longer time.

Recently, a lot of refugee and migrant families from all over the world faced a lot of struggles at the U.S immigrations. A lot of these refugees faced problems with the documentation because they could not understand the content of the legal documents. It is natural for people to understand information better in their home language. Therefore, an effective method of providing and accessing information in multiple languages is required.

In this study, in order to solve such problems, a multilingual information service was developed using Computer Vision and Bluetooth beacons. The information can be accessed from the user's smartphone in most of the languages. We focus on the smartphone's camera to 'see' information in multiple languages. We also use bluetooth beacons to 'push' information to smartphones in proximity. The user can then access the information in multiple languages. By using this method, we expect people visiting foreign countries to access information naturally in the same way they would in their home countries. This paper evaluates various options like GPS, NFC and RFID which could be used to provide information based on user's location. We then describe the required functions and configuration of the prototype system developed.

2 Background

Text is highly researched area in the computer vision application to model a smart self-learning system, which involve text associated with shop banners, highway and roadside sign boards or the text on local transport system, such a text provide significant clues about that environment. Text detection helps visually impaired and tourist to convey correct information in more understandable way to user.

As a method of accessing information on public signages, there are two ways of using a smart phone to find information in relevant languages; the user needs to search information by himself, then translate it relevant language to access necessary information. On the other hand, when the digital signage is used, the user can find and acquire information by walking in front of the digital signage without performing any special action. In this study, we explore ways to improve searching and accessing information by using smartphone camera and the method of information provision using the digital signage using iBeacons. This should make users who visit foreign countries get information without making special efforts.

2.1 Related Work

A study of multilingual digital signage using iBeacon communication was done, for the purpose of evaluating the technology in an university campus at Japan. [11] In the experiment, the authors installed a total of 25 iBeacon devices at some sightseeing spots, restaurants, souvenir shops, photo shops, etc. in Shirakawa-go, Japan. When vistors look at these heritage spots from the outside, it is difficult for them to understand whether it is a restaurant or a souvenir shop. By using this system, the tourists was able to obtain necessary information in their native languages when they enter each iBeacon area while walking in Shirakawa-go. The paper confirmed that this system can be used effectively for both Japanese and the foreigners [11]. In this system, it was possible to change the displayed language automatically when the user comes near to the digital signage, by using the mechanism of background communication of the iBeacon. **However, the study is dependent on installing and configuring bluetooth iBeacons for the user to be able to understand the signages.**

American MLB has been equipped with iBeacon. Its mobile application can provide navigation system, so the ticket in your hand will pop out automatically and your position will be indicated when you get close to ticket entrance. Several techniques such as the method of displaying information written in several languages in cyclic order, the method of displaying multilingual information simultaneously, and the method of switching displayed language using touch buttons, have been used. [6] When iBeacon is connected to the nearest iBeacon base station, the position information of base station will be gained and customers will be provided with high-quality service.[8] [19]

In India, a study of text detection and extraction based on Stroke Width Transform (SWT) and methodology to extract letters was conducted. [13] Major application focus were tourism industry and local transport, to help people to deal with different Indian languages which involve text associated with natural scenes in the local public places. Using SWT method, various Indian languages and their combination with English were detected. **However, this study does not translate the detected texts into the user's preferred languages. This is still a hindrance for foreigners to understand their environment.**

The problem of proximity estimation is found to be difficult in a variety of environment. Existing approaches such as Global Positioning System and Wi-Fi triangulations are insufficient to meet the requirements of accuracy and also requires high cost. A study was made using, Bluetooth which is commonly available in all smartphones to find the proximity over a shorter distance and provide an estimation model to determine the distance based on the Received Signal Strength Indicator (RSSI) values of the Bluetooth [12]. In existing system GPS is used to find the location, it won't work as accurately indoors and inside most commercial building areas. So the authors proposed a system to overcome the problem by using Bluetooth to Bluetooth proximity estimation. By using the signal strength of the Bluetooth device, estimated RSSI value is used to find out exact distance between the devices. This technique helps in tracking the location of nearby user. The study proposed the proximity estimation model by combining Bluetooth RSSI value. The results demonstrate that Bluetooth offers an effective mechanism that is accurate and power-efficient for measuring face-to-face proximity to in-

crease Bluetooth signal length [12]. **While this study does not focus on text detection or digital signage, this is useful for our study to determine the distance of the user/smartphone to the iBeacons**

Optical Character Recognition (OCR) is the electronic conversion of images into machine encoded text. It provides alphanumeric recognition of printed or handwritten characters. OCR has been an active topic of research in the recent past, and has wide applications in banking, healthcare, finance and education. According to the World Health Organization (WHO), around 285 million people around the world are estimated to be visually impaired, out of which 90% live in developing countries. Thus there is a pressing need to develop a system to provide information to the visually impaired. The authors proposed a camera based framework built on the Raspberry Pi, integrated with Image processing algorithms, OCR and Text-to-Speech (TTS) synthesis module [15]. The camera module is used to capture an image of the printed text, and the image was then subject to preprocessing before being fed into the OCR.[15] **This study, however does not focus on language translation or digital signage, we use the adopt image pre-processing techniques like binarization, de-noising, deskewing, segmentation and feature extraction in our research.**

In Taiwan, to encourage learning English, a study was carried out using the iBeacon's micro-positioning function to set the location in museum, restaurant, store, etc. When your mobile phone detects the information of iBeacon's location situation, you can learn by interacting with users' surrounding environment. [8] English obtained from the vocabulary performance includes the single words and expressions commonly used in daily life. With the listing narration and use of straightforward sentence structure and grammar, users can learn English conveniently and rapidly and apply it flexibly to real situations. In this way, users can learn English whenever and wherever possible to enhance English ability, achieve better effects with half efforts and gain language learning fun in the proposed mobile application. With the combination of the proposed application, English learning materials and iBeacon micro-location feature, learners can receive the surrounding related learning materials. In this manner, learners will have chance to build the connection between the learning materials and the environment which can enforce their memory to remember the learning concept. [8] It was found that the system

can raise the learning intent of the learners and may improve their learning effectiveness. **However, this study focusses on converting Chinese language to English based on the user's location. It does not support multiple languages and the system is entirely dependent on iBeacons.**

With the rapid increase in data and multimedia services, demand for positioning has increased especially in complex indoor environment which often needs to determine the location information of the mobile terminal. There is a lack of accuracy and robustness in current indoor positioning system. A study was made to design and implement a mobile-based indoor location system which has the mobile applications with the Bluetooth Low Energy technology based on the iBeacon. This study designs and implements an indoor positioning system based on iBeacon [9]. The authors adopted Gaussian filter and Unscented Kalman filter method to robustly extract strong signals from iBeacon device. With the extracted signals, the authors compared them with-in database. They were able to display data based on micro-location on the user's smartphone. It was found that the iBeacon approach has better performance compared with WiFi method [9]. **However, this study does not translate the detected texts into the user's preferred languages. This is still a hindrance for foreigners to understand their environment.**

2.2 Outline

With the development of Internet of things (IoT), the user and their location is closely interrelated. The outdoor positioning system based on satellite position (GPS) has matured enough that it provide users with the needs of high-precision positioning. However, GPS positioning satellite signals can only be received in outdoor environment, it difficult to meet the needs of indoor positioning [9]. So, there is an urgent demand for indoor positioning technology and providing micro-location-aware data.

In recent years, various types of emerging short- range wireless communication protocol and related products are presented in the market, where the standards of short-range wireless data communication includes ZigBee, WIFI, Bluetooth, NFC, RFID and Ultra Wideband (UWB) and other types

of technology. These technologies play an important role in their respective areas as they meet current application requirements. Let us evaluate some of the options we have to provide data based on micro-location.

2.2.1 GPS

The outdoor positioning system based on satellite position (GPS) provides users with the needs of high-precision positioning, accurate to 5 meters. However, GPS positioning satellite signals can only be received in outdoor environment. GPS fails to determine the user's location accurately inside a building or within a room [9]. Hence, we cannot use GPS to accurately provide users with micro-location aware data.

2.2.2 Wi-Fi

The coverage of radio wave is very wide, it can reach 100 meters, and achieve coverage with comprehensive. Also, the speed of network transmission is higher, it's useful for real-time interactive facet of the application. The Wi-Fi network construction has a low cost, easy to maintain and very suitable for use in mobile phones. However, the core drawbacks are data security and performance slightly less compared to Bluetooth [9]. It is also difficult to accurately triangulate the user's position inside the network.

2.2.3 Near Field Communication (NFC)

NFC is a set of short-range wireless technologies that enable two electronic devices, one of which is usually a portable device such as a smartphone, to establish communication by bringing them in close proximity. However, the range of NFC is less than 10 centimeters and it gets very expensive to install a lot of NFC tags. Thus, it is not feasible to use NFC in our study.

2.2.4 Bluetooth

Bluetooth is a radio technology which supports short distance communication from each other, the positioning of Bluetooth technology is the measuring of radio wave signal intensity values for targeting. The main advantage of Bluetooth indoor positioning technology is that, Bluetooth chips are small, easy to integrate in mobile phones and even in smaller devices. On the other hand, the shortcomings of Bluetooth technology are; more expensive

equipment, relatively high power consumption and lack of stability when the system in complex space environment [9].

2.2.5 iBeacon

The communication protocol used in iBeacon is the Bluetooth Low Energy (BLE), maximum Bluetooth version 4. 0 devices supports Bluetooth BLE version [9]. iBeacon communication frequency consumption is the 2.4GHz band, which is as fast as the Wi-Fi. iBeacon has proximity sensing technology of BLE, it can transfer Uniform Code of unique ID (UUID), APP intelligent terminal obtained the information about UUID and RSSI, and it can be converted into a physical location, which triggers location-aware applications. iBeacons are also low-cost devices and typically run for an year on a single coin CR2032 battery. [16] This makes it feasible for us to implement iBeacons into our study of providing data based on user's micro-location.

2.2.6 Comparison of indoor positioning methods

Below table presents the comparison between advantages and disadvantages of various indoor positioning systems [1]. It can be seen that iBeacon has advantages of low cost, long battery life, feasible accuracy and range. [17]

Also the fact that iBeacon technology is supported by most of the modern smartphones [9] makes it the most feasible choice to implement the multilingual signage system for our study.

Table 1: Comparison of popular indoor positioning methods

	Wi-Fi	. RFID	iBeacon
Cost	Medium	Medium	Low
Accuracy	Low	Low	High
Range	Far	Medium	Far
Advantages	Highly popular, far sensing range	Feasible accuracy and range	Low cost, high accuracy and range.
Disadvantages	Low accuracy, influenced by environment	Low popularity, difficult to maintain	likely to be influenced by environment.

3 Understanding iBeacon Technology

iBeacon is Apple's implementation of Bluetooth low-energy (BLE) wireless technology to create a different way of providing micro-location-based information and services to nearby smartphones supporting BLE. This allows mobile applications on a smartphone to determine when it has entered or left the region, along with an estimation of proximity to a beacon.

Bluetooth 4.0 is proposed according to the Bluetooth technology standard set by SIG to achieve two-way transmission in technology. Compared with traditional Bluetooth, Bluetooth 4.0 has the advantages of lower cost and lower energy. One button cell can make the Bluetooth Low Energy be operated for 1 to 2 years. [8]

iBeacon devices simply broadcasts its configured advertising packets in specific time intervals [11]. Apple has standardized the format for BLE Advertising. Under this format, an advertising packet consists of four main pieces of information.

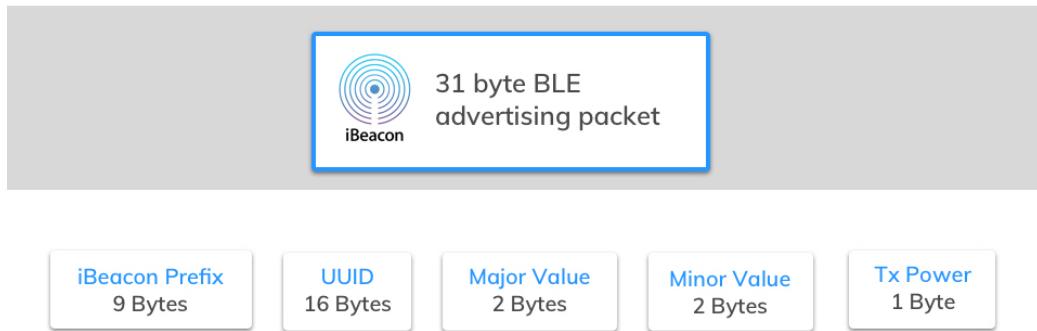


Figure 1: iBeacon packet format

Breaking down the iBeacon format, there are five parts: fixed iBeacon prefix; the proximity UUID: an identifier used to distinguish your Beacons from others, such as your stores or a museum; the major number can group

a related set of beacons in an organization, such as a floor in a store or a exhibition hall in a museum; the minor number can be used to identify individual beacons in a specific areas because each Beacon have a different minor number, by which we know where the customer is exactly. TX power, is used to determine how close you are to a beacon [7]. It can be presented either as rough information (immediate/far/out of range) or as a more precise measurement in meters.

UUID: This is a 16 byte string used to differentiate a large group of related beacons. For example, if National Park Services (NPS) maintained a network of beacons among all national parks, all national park beacons would share the same UUID. This allows NPS dedicated smartphone app to know which beacon advertisements come from NPS owned beacons. Example: E6FA79C7-804F-4742-863B-BD4D282ED9BA

Major: This is a 2 byte string used to distinguish a smaller subset of beacons within the larger group. For example, if NPS had fifty beacons in a particular national park, all fifty would have the same Major. This allows NPS to know exactly which park its visitor is in. Example: 1101

Minor: This is a 2 byte string meant to identify individual beacons. Keeping with the NPS example, a beacon at the front of the park would have its own unique Minor. This allows NPS's dedicated app to know exactly where the visitor is in the park. Example: 9901

Tx Power: This is used to determine proximity (distance) from the beacon. TX power is defined as the strength of the signal exactly 1 meter from the device. This has to be calibrated and hardcoded in advance. Devices can then use this as a baseline to give an estimate the distance between the beacon and the smartphone.

3.1 Existing applications implemented using iBeacon technology

- Control the shoppers' behaviors via recommending services automatically by iBeacons. In a store, the retailer can sense shoppers' interests

via the calculation of time a shopper in front of a specific item in the store, and then, the software in shopper's smartphone will show the other color or style of this item, or even let shopper access to the online shop to compare the price and style at real time [7].

- The retailers can gather shoppers behavior info, for instance, the shopper have ever seen the same goods in the other store of the same chain online or onsite with iBeacon technology [7]. And then, push the good coupons info at the store, meanwhile, show the shopper this goods in their online shop, so as to guide shoppers make purchase online or offline.
- iBeacon technology brings many chances for the retailers to keep broadcasting the items/events info automatically by smartphone application of users [7]; to tracing users interests or potential demands; to contextualize shoppers' purchase tendency.

3.2 The algorithm to calculate distance based on RSSI

The rate of signal strength will be unstable because of the volatility of RF signal in actual data collection method. So there was no longer accurate correspondence between the RSSI and distance. Consequently, we need to purify the sample data in order to reduce the relevant positioning errors.

We place iBeacon at some fixed position in the test environment before positioning, the mobile terminal detects the response of iBeacon signal strength, and we build RSSI vector set based on it, this set represents the RSSI vector is $R=(R_1, R_2, \dots, R_i, \dots, R_P)$, where i is the signal strength of i -node RSSI, P is the total number of iBeacon.

We then repeatedly measured RSSI and averaging the values, the average value as the iBeacon characteristic value $r=(r_1, r_2, \dots, r_i, \dots, r_m)$, the formula is as follows:

$$RSSI = 1/m \sum_{i=1}^m RSSI_i \quad (1)$$

where m is the total number of collected RSSI vector at the coordinate points. This approach would correct the rather unstable distance of the beacon from the user.

4 Google Vision API

Google Cloud Vision is an image recognition technology that allows us to remotely process the content of an image and to retrieve its main features. By using specialized Representational State Transfer (REST), called Google Cloud Vision API, developers exploit such a technology within their own applications [2]. By using Google’s Cloud Platform to compute the content of an image through advanced machine learning processes, this solution allows developers to extract some relevant information from visual data, including image labeling, face and landmark detection, optical character recognition (OCR), and tagging of explicit content. It is possible to interact with Google’s Cloud Vision platform by using specialized REST API, called Google Cloud Vision API [2].

We propose to exploit such cloud-based software resources in order to achieve a system for people with understand signages and their environment in general. In particular for users who are visiting foreign countries, our solution may help them to interact with the environment and the things around them. In this paper, we focus on the OCR functionality of the Google Cloud Vision. We also discuss the implementation and integration of the Google Vision into our software solution, a mobile application we developed. The smartphone requires an Internet connection to submit the captured images to Google’s cloud platform. The response from platform is processed and displayed on the user’s smartphone.

The algorithm shown in Figure 2 has been implemented in Javascript. In particular, here the request for Google Cloud Vision includes the TEXT DETECTION feature. A sample request and response is shown below:

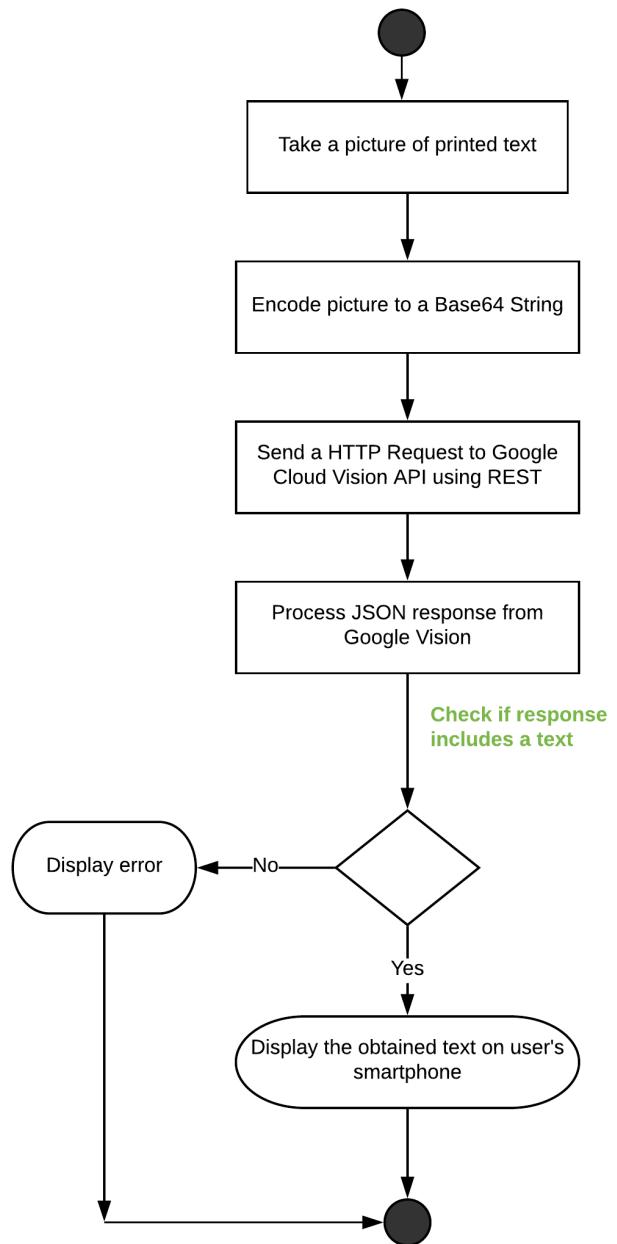


Figure 2: UML activity diagram for Text Detection using Google Vision API

4.1 Sample Request

Let us consider the image shown in Figure 3 as the source of the image. We use the Algorithm mentioned in Figure 2 to extract the text from the image.



Figure 3: Example image for testing Google Vision API

We convert this image to a base64 encoded string using Javascript and specify in the headers of the REST API to use 'Text Detection' feature of Google's Cloud Vision API. A sample request is as shown below:

```
{  
    "requests": [  
        {"image": {  
            "content": "base64_encoded_string_of_image"  
        }  
    ]  
}
```

```
        "features": [{"type": "TEXT_DETECTION",  
          "maxResults": 1}]}]  
}
```

4.2 Sample Response

```
{  
  "responses": [  
    {"textAnnotations": [  
      {"locale": "en",  
       "description": "Lorem Ipsum",  
       "boundingPoly": {  
         "vertices": [  
           {"x": 327, "y": 412}, {"x": 1158, "y": 412},  
           {"x": 1158, "y": 680},  
           {"x": 327, "y": 680}]]}]}]
```

In this section, we discussed the capabilities of Google Cloud Vision API and how we implement it into our system.

5 Google Cloud Translation API

The Translation API provides a simple programmatic interface for translating an arbitrary string into any supported language using state-of-the-art Neural Machine Translation. It is highly responsive, so websites and applications can integrate with Translation API for fast, dynamic translation of source text from the source language to a target language (such as French to English). Language detection is also available in cases where the source language is unknown. The underlying technology is updated constantly to include improvements from Google research teams, which results in better translations and new languages and language pairs.

Translation API supports more than one hundred different languages, from Afrikaans to Zulu. Used in combination, this enables translation between thousands of language pairs. Translation API can be accessed using REST APIs. We extract text from the printed signages using Google Cloud Vision API, we then send the text to Google Translate API and convert it into the user's desired language.

The figure 4 describes the process of language translation in our system. We obtain the text from the printed media using Google Vision API as mentioned in section 4. We then pass the obtained text to Google Cloud Translation API with the preferred language in the headers of REST API. The response from the Google Translate is then displayed in the user's smartphone. A sample request and response is shown below:

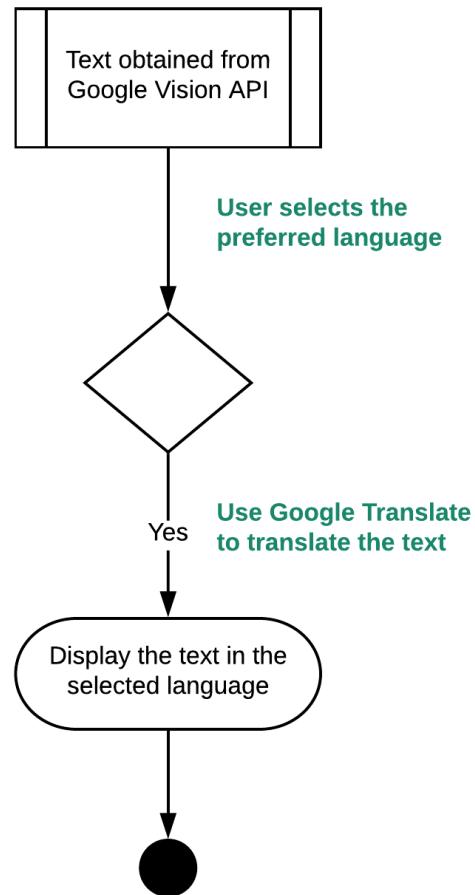


Figure 4: UML activity diagram using Google Translation API

5.1 Sample Request

Below code segment shows how to use Google Cloud Translate API using REST APIs using Javascript. Developers should replace their own Google API Key at API-KEY-HERE. The target language to be translated is passed under 'target' parameter of the REST API.

We use javascript to interact with Google Translation API as follows:

```

fetch('https://translation.googleapis.com/language/
translate/v2?key=API_KEY_HERE', {
  method: 'POST',
  headers: {
    'Accept': 'application/json',
    'Content-Type': 'application/json',
  },
  body: JSON.stringify({
    q: "text_to_be_translated",
    target: "target_language"
  })
})
.then(data => data.json())
.then(res => this.cleanText(res))
.catch(err => (err))
}

```

5.2 Sample Response

Google processes the requests and responds with a JSON object containing the translated text. A sample response is shown below:

```
{
  "data": {
    "translations": [
      {
        "translatedText": "translated_text_here"
      }
    ]
  }
}
```

6 Algorithm

The below pseudo-codes show the algorithms involved in our system. The system constructed for the study runs both the below algorithms in the background. For convenience, we have divided the algorithm into 2 sections, one for digital signage and one for printed signages.

Algorithm 1 Pseudocode for digital signages

```
1: procedure SCAN FOR iBEACONS
2:   if iBeacons found then
3:     loop:
4:       acquire RSSI info of beacon
5:       calculate and display distance of beacon
6:       query beacon parameters with server database
7:       if data exists then
8:         fetch and display data
9:       procedure USER SELECTS PREFERRED LANGUAGE
10:        fetch selected language
11:        send the text and selected language to Google translation
12:        fetch and display text in user's preferred language
```

Algorithm 1 shows the pseudo-code for scanning iBeacons. The system built in the study scans for beacons in the background. When the beacons are found, the mobile application reads the parameters of each beacon and queries the database in the server with the parameters. If data is found in the server, the mobile application processes the data and displays appropriate signage information. The user can then choose to translate the data into his preferred language using Google Translation.

Algorithm 2 shows the pseudo-code for extracting text from printed signages and then translating them to preferred language. The system listens to the event of image capture from smartphone camera or image upload from the camera roll. On this event, the system converts the image into a base64 encoded string and communicates with Google Cloud Vision using REST method. If the server is successful in extracting text from the image, it

Algorithm 2 Pseudocode for printed signages

```
1: procedure USER CAPTURES AN IMAGE OF SIGNAGE
2:   convert image to Base64 encoded string
3:   send the image to Google cloud vision
4:   if text returned then
5:     parse the response and display text
6:   procedure USER SELECTS PREFERRED LANGUAGE
7:     fetch selected language
8:     send the text and selected language to Google translation
9:     fetch and display text in user's preferred language
```

responds with a JSON file containing the extracted text. The mobile application then processes the data and displays the extracted text. The user can then choose to translate the data into his preferred language. On selecting a language, the system then sends both the extracted text and selected language to Google Translation using REST method. On successful translation, the system then displays the information in user's selected language on the mobile application.

7 System Design

In this study, a prototype of multilingual information service system was designed and constructed. 3 beacons was configured and placed next to each of the 3 signage in a park. These beacons continuously broadcast their UUID, major and Minor. A mobile application is developed to listen to these beacons. The mobile application leverages the smartphone's bluetooth technology to listen to the beacons.

The relevant information of the signage for each beacon is stored in a database in the cloud. Once the mobile application detects a beacon, it makes a call to the cloud, passing the detected beacon configuration to get the appropriate information of the signage using REST APIs. The information returned from the cloud is then translated to required languages using the Google Translate API and displayed on the user's smartphone.

The below figure 5 gives an overview of different processes in our system.

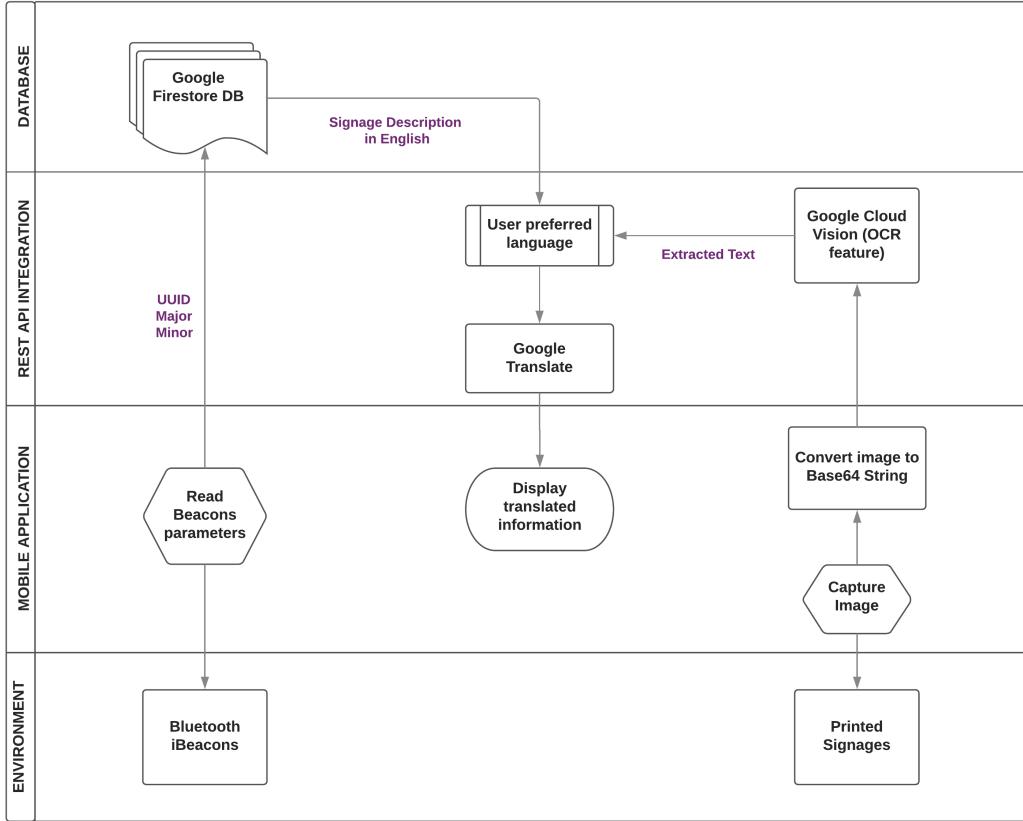


Figure 5: Process Flow Diagram

7.1 Environment

To evaluate the system, we constructed a combination of digital signages and printed signages and placed them in the university campus. The students, including foreign students were handed out iPhones running the mobile application and were requested to use our system.

7.1.1 Digital Signages

The digital signage was set up using the Estimote 2018 iBeacons. Each iBeacon is mounted on individual signages in the campus. The beacons are powered by individual CR2032 lithium-ion coin batteries. The users used the iPhone 6s in which the application program that reacts with the

iBeacon signals. In this study, we used the iBeacons from www.estimote.com. The parameters of the beacons, which are discussed in the section 3 can be configured only by the administrator of the mobile application. To evaluate the system, we use 3 beacons. Below are the configurations of each beacons used in the study.

UUID: E6FA79C7–804F–4742–863B–BD4D282ED9BA

Major: 1111

Minor: 9991

UUID: E6FA79C7–804F–4742–863B–BD4D282ED9BA

Major: 1111

Minor: 9992

UUID: E6FA79C7–804F–4742–863B–BD4D282ED9BA

Major: 1111

Minor: 9993

7.1.2 Printed Signages

To evaluate our solution of translating the signages without the iBeacons, we printed out commonly used signages in multiple languages and placed them around the university campus. The students were handed out iPhone 6s, running the same mobile application and were requested to use the system to understand the information on the signages.

7.2 Mobile application

To discover the beacons from the smartphone, we have designed and developed a mobile application. This mobile application is built using React native and Javascript. The mobile application uses Google Cloud Vision to detect the printed text using the smartphone camera. The Google Cloud Vision API detects the printed information using OCR feature as mentioned in section 4. This text can be converted to any language using Google Cloud Translation API, discussed in section 5. The mobile application interacts

with these APIs using REST method To improve the efficiency and user-experience, we used both iBeacon monitoring and ranging to discover and interact with the beacons.

7.2.1 Background Monitoring

Beacon monitoring is similar to geofence, i.e., a virtual barrier that is usually defined using a set of geographic coordinates. In case of iBeacon, the area is defined by the range of one or more beacons. This allows more granularity and precision than regular geofencing. Moving in and out of the area it encloses triggers 'enter' and 'exit' events, which the app can react to. The smartphone will keep listening for those beacons at all times even if the application is not running in the foreground, and even if the iPhone/iPad is locked or rebooted. Once an 'enter' or 'exit' happens, iOS will launch the app into the background (if needed) and let it execute some code for a few seconds to handle the event.

7.2.2 Beacon Ranging

Ranging actively scans for any nearby beacons on the foreground and delivers results to you every second. With monitoring, our app will be notified whenever the user enters and exits the terminal. Ranging for the exact same region, we'll instead get a full list of matching beacons currently in range - complete with their UUID, major, and minor values.

7.2.3 Development stack

The mobile application was developed after evaluating all the major development tools and technology present in the current market. The main goal is to develop the system, keeping performance and scalability in mind. Also, we want to make sure the system works efficiently on both iOS and Android platforms. Below is the development stack we used to develop the mobile application:

- **React Native:** To develop the front end of the mobile application. The advantage of using React Native is it helps us to develop for both Android and iOS with the same codebase. Also, React Native is efficient in terms of performance.

- **JavaScript:** We used Javascript to interact with the web-services such as Google Vision and Google Translate. It makes it efficient to convert the captured images into Base64 encoded string and pass it in the headers of the request. Also, javascript makes it easier to process/parse the JSON response from the web-service and display it on the front-end using React Native. We use Javascript also to communicate with the Google Firestore database to query and retrieve data.
- **Camera:** To access the smartphone camera, we included a library called 'react-native-camera'. This extends the functionality of camera in terms of fragments and hence makes it easier to listen to the camera events and implement our logic where necessary.
- **Bluetooth:** Our system needs access smartphone's bluetooth module to discover the iBeacons. To check the state of Bluetooth, we included 'react-native-bluetooth-state' library. This helps us determine the bluetooth status of the smartphone, whether 'on' or 'off' and display appropriate messages to the user.
- **Beacon Manager:** To discover iBeacons from the smartphone, we integrated 'react-native-beacons-manager' library into React Native. This library helps us scan for the beacons and listen to changes of the beacon signal strength. The library also supports 'background ranging' and 'foreground monitoring' of the beacons, meaning it helps us scan for the beacons even when the application is running in background.
- **Xcode:** To build and run the application on the iOS device, we need an Integrated Development Environment (IDE). Xcode is an IDE that can be used to run the developed mobile application on the iOS device. The source code for the implementation could be found at [3]

7.2.4 Screenshots



Figure 6: Home page
Camera and digital signages around



Figure 7: Text extraction
Text extracted from the picture

The above figures show the screenshots of the mobile application. Figure 6 shows the Landing screen of the application. The user sees this screen on opening the application. The Home page has the following functionalities:

- Capture image of a physical signage
- Scan for digital signages around. If found, display the image, title and the distance of the digital signage from the user.
- The user can also upload an image from camera roll of the smartphone and extract text from it.

- The user can choose to type a text and convert it into any language

Figure 7 shows the text extracted from the image captured on figure 6. This screen is presented to the user right after capturing an image of a physical signage. We convert the captured image into Base64 string and pass it to Google Cloud Vision using REST and extract the text from the image. The screen on 7 has the below functionalities:

- Display the extracted text from the image
- The user can choose to translate the extracted text to any of the supported language.

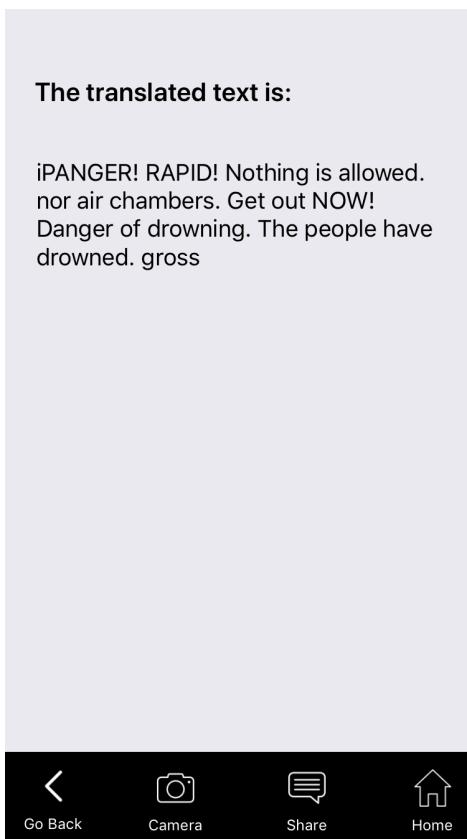


Figure 8: Translation
Extracted text translated to Spanish

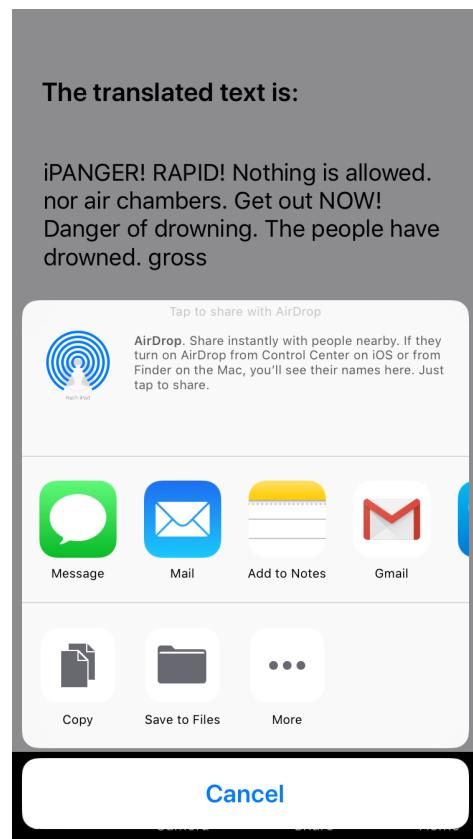


Figure 9: Share text feature
Options to share the translated text

Figure 8 shows the text translated into Spanish. This is the same text extracted from figure 6. The extracted text is shown on 7. This screen is presented to the user after the user chooses a language and taps on the 'GO' button. We listen to the event, capture the language selected and send a request to Google Translation API using web-services. The screen on figure 8 has the below functionalities:

- Display the translated text in user's preferred language.
- Share the translated text using Share button.
- Capture another image of a physical signage.
- Go back to landing page.

We strongly feel that there should be an option to share the translated text in any form. Figure 9 shows the Share feature we integrated so the user can record the translated text and share them with friends and family who might need the information. In our system, the user can choose from the multiple options to share the translated text with. The default options of the iOS include e-mail application, messaging application, notes application etc.

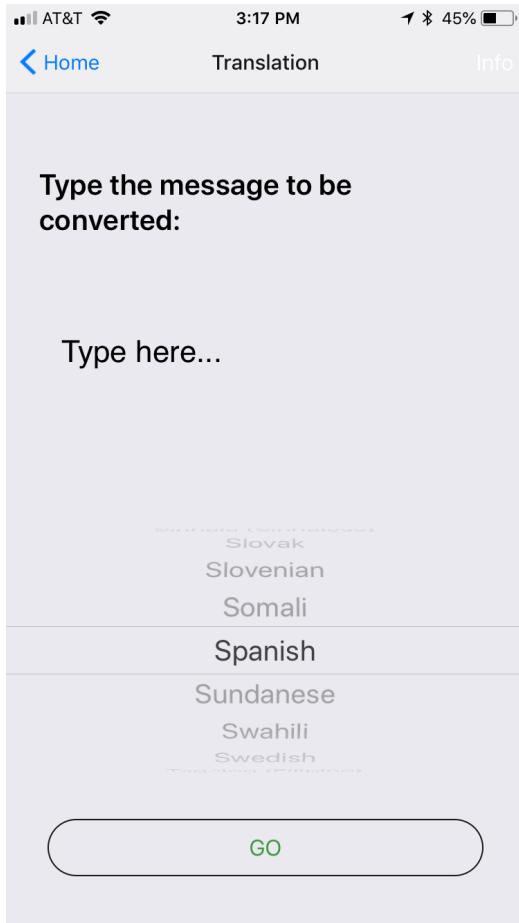


Figure 10: Type and translate text feature

Since our primary users of our application are foreign visitors, we feel there is a strong need for them to communicate day-to-day for critical information in their native language. To achieve this in our system, we implemented a 'type and translate' feature. Figure 10 shows the screenshot of the 'type and translate' feature. The user can type in text in any language and choose to convert it into any of the supported languages.



Figure 11: Digital Signage Details about the signage

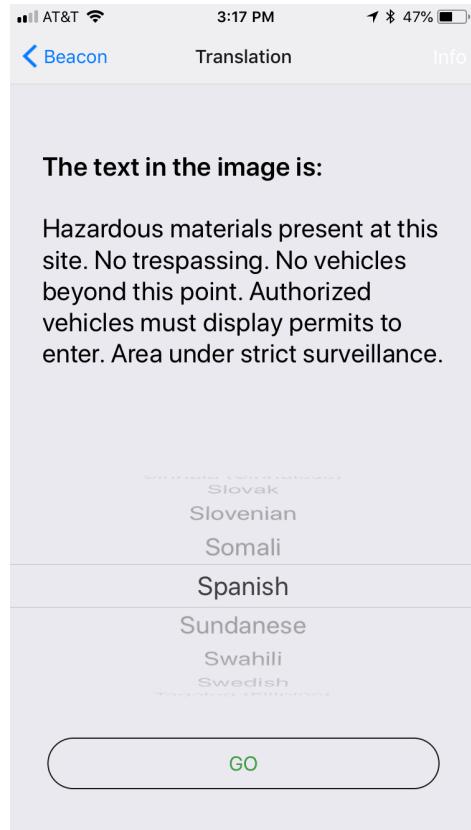


Figure 12: Translate feature Translate text to other languages

Figure 11 shows more details of the digital signages which are listed on the landing page shown in 6. The user lands on this screen after tapping on any of the digital signage thumbnail. When the user taps on a particular signage, the system screen fetches the pre-configured information and media from the database and displays them on this screen. This screen displays the image of the signage, the title and the description of the signage in English. This component has the below functionalities:

- Fetch and display the appropriate data (media, title and description) from the database.
- Provide the user with the option of translating the information into supported languages.

Figure 12 is shown when the user taps on 'Translate' button from figure 11. The information of the digital signage is passed to this screen. Now, the user can translate the text into any supported language. This component is capable of the below functionalities:

- Display the beacon information.
- Provide the user with the option of translating the information into supported languages.

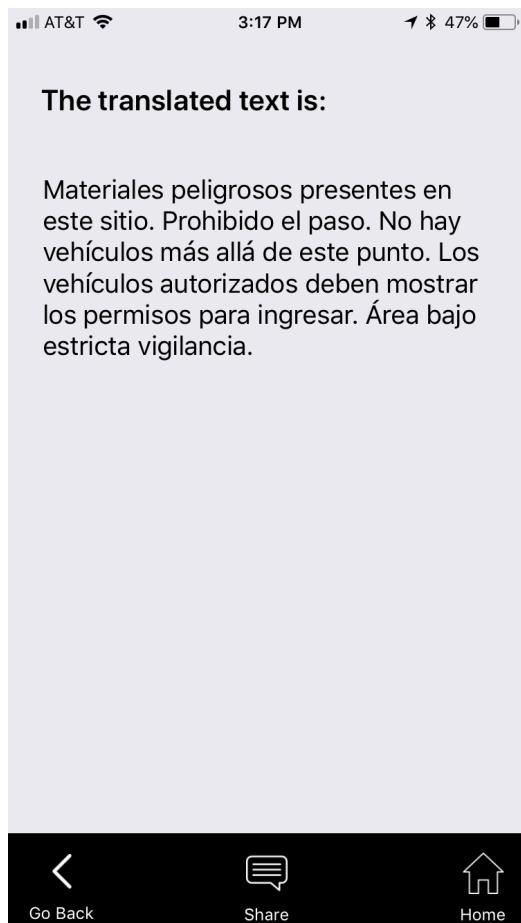


Figure 13: Text translated to Spanish language

Figure 13 is shown when the user selects a language and taps on 'Go' button from figure 12. The information of the digital signage is passed to this screen. Now, the user can translate the text into any supported language. This component is capable of the below functionalities:

- Display the translated text in user's preferred language.
- Share the translated information.
- Go back to landing page.

7.3 REST API Integration

The mobile application developed communicates with the web-services such as Google Cloud Vision and Google Translation using Representational State Transfer (REST). REST is an architectural style that defines a set of constraints to be used for creating web services [18]. Web services that conform to the REST architectural style, or RESTful web services, provide interoperability between computer systems on the Internet.

By using a stateless protocol and standard operations, REST systems aim for fast performance, reliability, and the ability to grow, by re-using components that can be managed and updated without affecting the system as a whole, even while it is running. In our system, we interact with Google Vision and Translation web-services through REST method. The response from these web-services is processed in the mobile application and appropriate messages are shown. A sample request and response for each web-services is shown in Sections 4 and 5.

7.4 Database

To store the information of each signage in a database, we chose Google Cloud Firestore. Cloud Firestore is a flexible, scalable database for mobile, web, and server development from Firebase and Google Cloud Platform [4]. It keeps the data in sync across client apps through realtime listeners and offers offline support for mobile and web so we can build responsive apps that work regardless of network latency or Internet connectivity [4]. Google Firestore is free of cost for less than 50 simultaneous connections to the database, which is a right fit for the study.

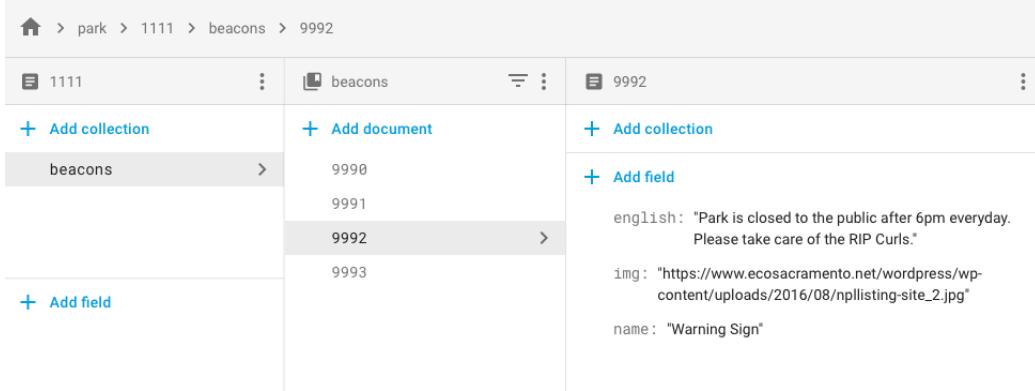


Figure 14: Database Schema

We designed the database schema with a goal to keep it as simple as possible so it can be scalable in the future. We organize the database schema as follows:

- **Park:** Root of the database. All the beacons with the specified UUID falls under this bucket.
- **1111:** This is the level 1 of the database. This is essentially the 'major' parameter of the iBeacons. Since all the 3 beacons we configured have the same 'major' value, we just have one collection of level 1. This level of database makes it possible for our mobile application to differentiate the sections of the park. This allows the administrator to configure beacon parameters according to sections of the park.
- **beacons:** This is the level 2 of the database schema. We now have identified which park the beacons are present in and which section of the park the beacons are present in using the above two levels. 'beacons' section of the database serves as the collection of beacons under a particular section of the park. They contain all the information about the individual beacons present in discovered section.
- **9992:** This is the last level of the designed database schema. '9992' refers to the 'minor' value of the iBeacon. Each iBeacon will have its own unique 'minor' value. This helps us to query for the exact beacon data. Each entry at this level have the below fields and values:

'name': The name of the signage.

'img': Contains the URL of a picture of the signage.

'english' : This field contains the description of the signage, in English.

8 Designing the User Experience

Applying UI Design principles and guidelines are very important for developing a mobile-based learning application. By observing user interface design principles and implementing them in the development process, it can be helpful for users to improve usability and grasp critical information as displayed in the application.

For this study, set of User Interface Design (UID) principles is considered [14] for designing and developing the mobile-based multi-lingual signage application. Below are the set of UIDs adopted:

- Navigation should be simple and clear from a page to any particular section. In short, the navigation should be consistent throughout all pages in an application.
- Complex navigation needs to be avoided by the developer on the application.
- Reduce scrolling frequently.
- Application should be user-friendly and allow users to understand how the application works intuitively.
- Similar actions and information need to be located in similar positions. For instance, similar buttons must be found in similar positions for the whole pages of the application.
- Flexibility of the display is the significant property for usability of interface design.
- Providing necessary information only. Unnecessary information confuses the learners and decreases their performance.
- Decreasing textual information and increasing information which is provided in graphical media to minimize learners' cognitive load and motivate them to keep using the application.

From the psychological point of view, interface design can be divided into feeling (visual, auditory and tactile etc) and emotional levels. User interface

design is an important part of on-screen applications. Interface design is a complex engineering which have different disciplines in, cognitive psychology, design, linguistics, etc. [5]. There are three main principles of User interface design: Intuitive user interface, Reduce the burden of user memory, maintain the consistency of the interface.

We are going to carefully design the User Interface (UI) and User Experience (UX) based on the best practices for Human Computer Interaction (HCI). Below sections describe the concepts behind designing the UI and UX.

8.1 User Analysis

Below are the different types of users we expect to use our system:

- **Primary Users:** People who visit foreign countries either for tourism or business. Usually, the signages are displayed in local languages. This is a hindrance to visitors. They would want to understand their surroundings.
- **Secondary Users:** Visitors of parks and exhibits who would like to know more about their surroundings in their native language. The application will scan for digital signages and present relevant information once the user is in proximity.
- **Tertiary Users:** Tertiary users are the users who interact with the translated information. They may be the native people interacting with the visitors.

8.1.1 Example Persona

This section describes an example persona of a primary user:

Rebecca is a freshman in the Environmental Sciences program. She is 22 years old, from Arlington, VA. Throughout her high school years, she has been interested in preserving the environment, and has been involved in numerous recycling and conservation activities. She has had various international travel experiences, including a trip to Japan for a youth group ministry project.

Rebecca lives on campus in the dorm she shares with two other college students. She is very comfortable with technology and uses Twitter, Facebook, and email to communicate with friends and family. Rebecca believes in technology to reach out to people and spread knowledge about Amazon rainforests through images and icons. Rebecca is motivated to participate in a Peruvian conservation project during the summer following her high school graduation. She is the current Vice President of the Environmental Society.

During her foreign travels with the society, especially to the national parks, Rebecca likes to understand more about the environment and always makes sure there is enough signages so people are aware of the rules and do not pollute the environment.

8.2 Task Analysis

In this section, we identify what the user is trying to achieve using our system. We also identify the tasks that need to be supported in order for the user to achieve his goal. Below are the list of tasks the system needs to support.

- The system needs to support taking pictures of the signage and extract the printed text.
- The system needs to translate text into the user's preferred language.
- The application should remember the selected language and display the translated text into the preferred language.
- The application should scan for digital signages and display the distance of the user from the signage.
- The application should fetch the appropriate information from the database and display the same.
- The administrator of the system should be able to configure the beacons and change the fields of the database as well.

8.3 Context Analysis

In this section, we discuss where the developed system can be used. We also discuss the social and environmental conditions. Below are the environments our system could be used:

- In parks and exhibits that involve multiple signages.
- Museums, where there is a need for foreign visitors to understand the signages in their native language.
- In shopping malls, so visitors can understand critical information and navigate easily.

8.4 Design Principles

The user experience was designed according to the design principles of Human Computer Interaction (HCI). In this section, we discuss the various design principles incorporated in designing the UX.

8.4.1 Affordance

According to Donal Norman (1988) an affordance is the design aspect of an object which suggest how the object should be used; a visual clue to its function and use [10]. Affordance is a property in which the physical characteristics of an object influence its function.

In the mobile application developed, the UI of the Landing page (shown on opening the application) is divided into 2 sections. About 3/4th of the screen shows the camera view, which means the application is ready to capture a picture of printed signages and extract text from it using Google Cloud Vision. The rest of the screen shows the nearby digital signages. The application also shows the distance between the digital signage and the user.

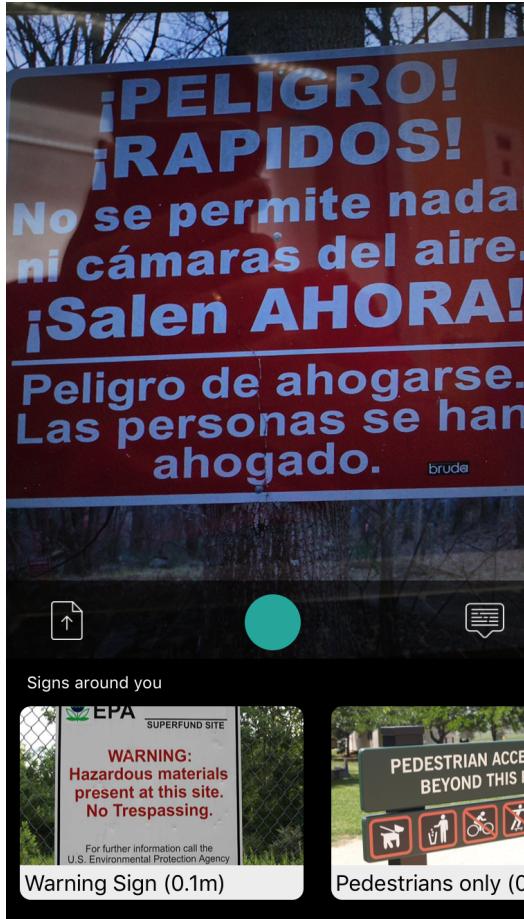


Figure 15: Affordance design principle

8.4.2 Visibility

The usability is improved when its status and methods of use are clearly visible. The more visible functions are, the more likely users will be able to know what to do next.[10] In contrast, when functions are 'out of sight,' it makes them more difficult to find and know how to use.

In the mobile application developed, the capture picture button has a green color, so as to invite the user to click the button. The camera functionality also has an 'upload image' icon and 'type and translate' icon present upfront. There is no ambiguity in the user interface. The digital signage section also

lists the nearby digital signages along with the title and distance from the user. This improves the visibility of all the features of the app without the user having to scroll down.

8.4.3 Constraint

The design concept of constraining refers to determining ways of restricting the kind of user interaction that can take place at a given moment [10]. This helps the application to do what it is exactly supposed to do and eliminates many error scenarios. This is helpful for the stable functioning of the application.

In the system we built, we establish constraints by only listing the digital signages around. If there are no digital signages around, 'No digital signages around' message is shown to the user. Also, the system is constrained to capture/upload only one image and use web-services to extract text from it. This helps the application function without many errors.

8.4.4 Consistency

This refers to designing interfaces to have similar operations and use similar elements for achieving similar tasks [10]. In particular, a consistent interface is one that follows rules, such as using the same operation to select all objects. This helps the user intuitively learn and use the application without having to spend more time on it [10].

In our study, we have designed the UI using the modules approach. That is, whether the user chooses to capture an image or upload from the camera roll, the layout is the same. In the digital signages section, all the signages are listed consistently and the whole thumbnail is clickable. Each click takes the user to a screen with more details about the signage. This page is designed consistently. This helps the user to understand our application faster.

8.4.5 Feedback

Feedback is about sending back information about what action has been done and what has been accomplished, allowing the person to continue with the activity [10]. This helps the user to understand there is a background

process running and at the same time, the user is constrained from starting another process or quitting the current process.

In the application developed, the user is shown 'loading' icon when a background process is running, for example, when an image is being uploaded through web-services. This helps the user understand that there is a process running in the background and also helps the user to understand the application better. Also, appropriate error messages are shown at the appropriate places to help user understand about the action being carried out [10].



Figure 16: Feedback

9 Case study

We conducted a small scale evaluation of the multilingual digital signage system as mentioned in section 7. The purpose of this evaluation is to study the feasibility of using the inexpensive iBeacons along with computer vision features of the system. We set up a combination of digital and printed signages in a room. We set up three iBeacons along with two printed signages and measuring points in a room, approximately 45m x 35m.

The iBeacons used in this experiment were Gimbal Series 10 beacons, distributed by Gimbal. For reception, we used installed the developed mobile application on iPhone 6s, running iOS version 11.3. We also setup two printed signages with information in Spanish language and placed them around the library, at least 30 meters apart.

9.1 Digital Signages

These include the signages with iBeacons installed. There are two interactions during the discovery of nearby digital signages from the mobile application. The first part is interaction between iBeacons and the user. iBeacon will broadcast signals at a certain interval, if the smartphone detects and locates the iBeacons around it, the application can handle the iBeacon event based on proximity location information, including major ID, minor ID and distance. The second part is interaction between the mobile application and database. During this interaction, the application should fetch respective information and display on the mobile application.

9.1.1 Test case 1

With the above mentioned setup of the signages inside the campus library, we use the mobile application to evaluate the feasibility of the proposed system. Each signage is at least 20 meters away from each other. We downloaded the developed mobile application on to the iPhone 6s and walked towards the room.

The mobile application picked up the closest digital signage when the user was 17.4 meters away. The other digital signages were listed at the distance of 22.3 meters and 27 meters away. On tapping each thumbnail of signage

Table 2: Observations from test case 1

	Data
Discovered	27.4 m away
Fetched signage data	Yes
Text translated	Yes
Error in text translated	0%

from the application, it fetched and displayed relevant signage information from the database. We were also able to translate the signage description from English to Spanish and the translation was 100% correct.

9.1.2 Test case 2

For this test case, we setup the digital signages closer than the previous test case. The aim of this test case is to evaluate for inference in beacon signals. Each signage was installed 15 meters away from each other. We ran the developed mobile application on the iPhone 6s and walked towards the room.

The mobile application picked up the closest digital signage when the user was 23.4 meters away. The other digital signages were listed at the distance of 26.3 meters and 35.7 meters away. On tapping each thumbnail of signage from the application, it fetched and displayed relevant signage information from the database. We were also able to translate the signage description from English to Spanish and the translation was 100% correct.

This proves that the digital signage system can be set up in close proximity to each other and the system will not face any inference from the crowded iBeacon signals.

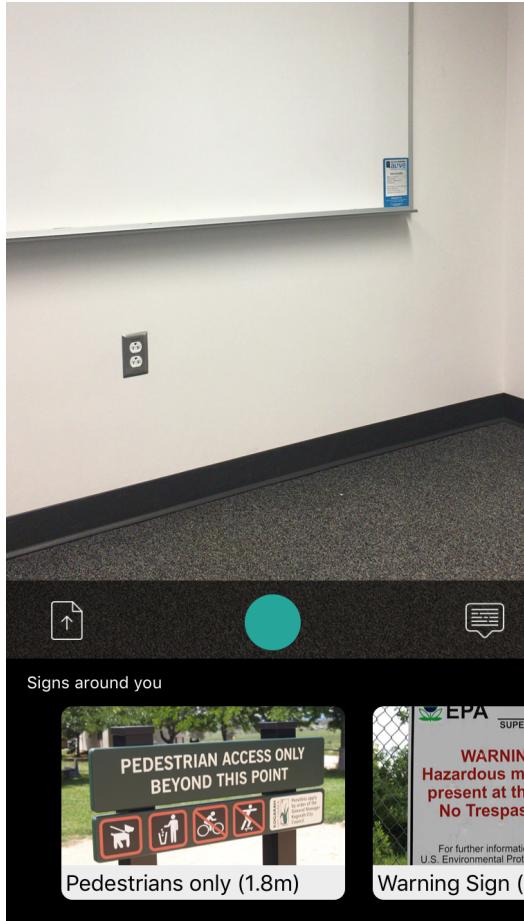


Figure 17: Digital signage discovered at 1.8 meters away

9.2 Printed Signages

The other scenario we need to evaluate is the multilingual translation of printed signages (signages without iBeacons installed). When the user clicks an image of a signage, the mobile application should be able to extract text and translate it to the user's preferred language without missing critical information. We evaluate the feasibility of the system in the following scenarios.

Table 3: Observations from test case 1

	Data
Discovered	27.4 m away
Fetched signage data	Yes
Text translated	Yes
Error in text translated	0%

9.2.1 Test case 3

We use the same mobile application used in test cases 1 and 2 to evaluate the feasibility of the system to extract text and translate it from Spanish to English. We positioned the mobile application at a distance of 35 feet from the signage.

To evaluate the text extraction and translation from printed signage, we clicked a picture of the printed signage. The application extracted relevant text from the printed signage. The application also supported the translation of the extracted text to Spanish language and the translation was 100% correct.

We noticed an 0% error in the text extraction and text translation from Spanish to English. Below figures 18, 19 and 20 show the flow of the text extraction and translation for printed signage 1.



Figure 18: Printed Signage 1
Signage in Spanish language

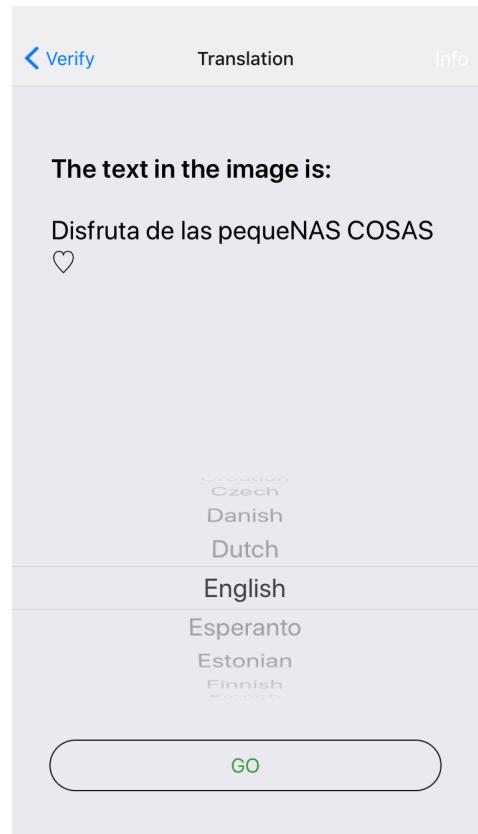


Figure 19: OCR feature
Text extracted from the signage

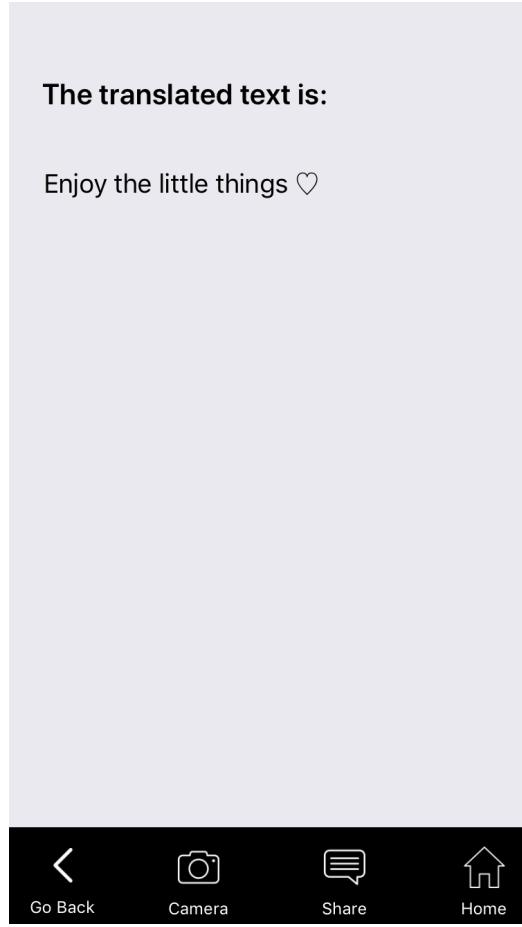


Figure 20: Text translated to English

9.2.2 Test case 4

For this test case, we use a printed signage with information in Spanish language and use our system to extract the text and translate it to English. The aim of this test case is to evaluate the text extraction and translation of different languages.

To evaluate the text extraction and translation from a Spanish printed signage, we clicked a picture of the printed signage. The application extracted relevant text from the printed signage. The application also supported the

Table 4: Observations from test case 3

	Data
Image captured from	-
Text extracted	-
Text translated	Yes
Error in text extraction	0%
Error in text translated	0%

translation of the extracted Spanish text to English language and the translation was 91% correct.

We noticed an 8% error in the text extraction feature and thereby the same error while translating the text to English. Below figures 21, 22 and 23 show the flow of the text extraction and translation for signage 2.

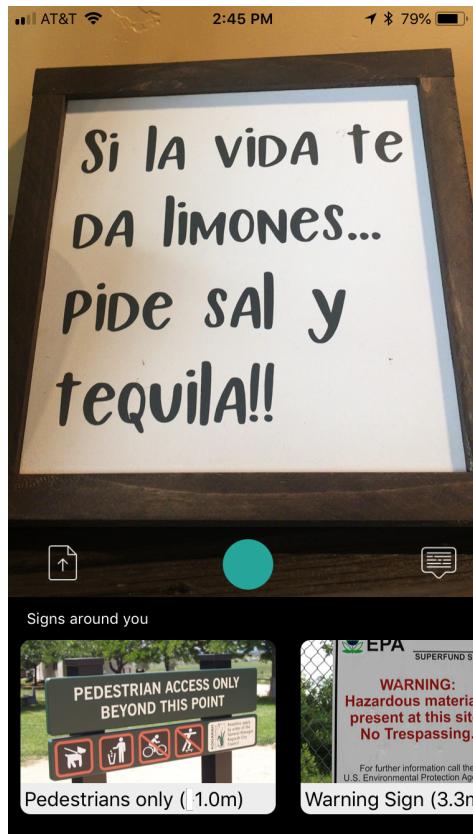


Figure 21: Printed Signage 1
Signage 2 in Spanish language



Figure 22: OCR feature
Text extracted from the signage 2

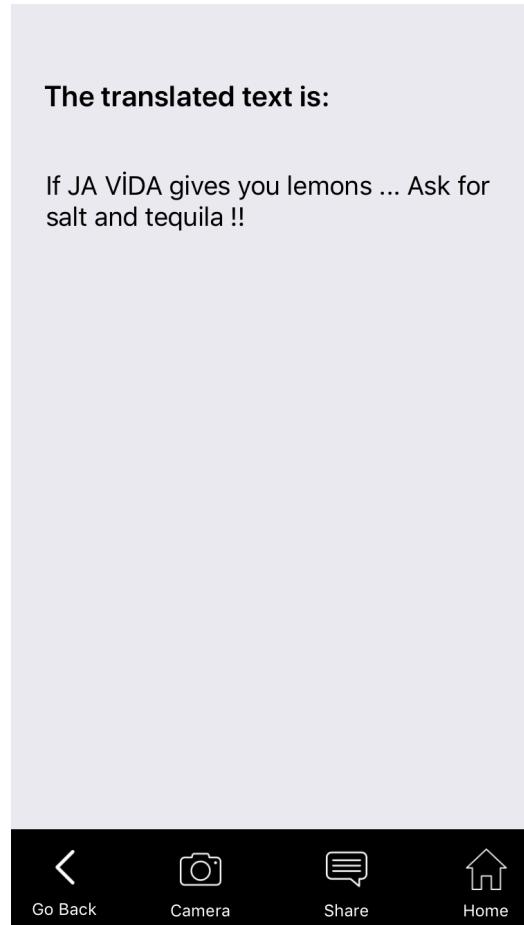


Figure 23: Text translated to English

9.3 Testing environment

Testing conditions	
Smartphone	iPhone 6s
iOS version	11.3
Internet	LTE
iBeacons	Gimbal Series 10

Table 5: Observations from test case 4

	Data
Image captured from	-
Text extracted	-
Text translated	Yes
Error in text extraction	0%
Error in text translated	0%

10 Conclusion and future work

We studied the feasibility of using the combination of inexpensive iBeacons and computer vision to construct a system of multilingual digital signages. We conducted a crude experiment using an actual room to estimate the positioning accuracy and accuracy of language translation.

- Evaluation experiment proves that the system is helpful to understand all kinds of signages in foreign countries.
- The construction of low-cost digital signage system proves to be effective in understanding the signages better for a foreign visitor
- The usability goals of the mobile application and the system as a whole is met.
- Determining the user's position in a network of beacon, thereby implementing indoor navigation.
- Translating the whole signage, while maintaining the format and graphics of the signage.

References

- [1] M. S. Aman, C. D. Quint, A. Abdalgawad, and K. Yelamarthi. Sensing and classifying indoor environments: An iot based portable tour guide system. In *2017 IEEE Sensors Applications Symposium (SAS)*, pages 1–6, March 2017.
- [2] Maria Fazio Massimo Villari Davide Mulfari, Antonio Celesti and Antonio Puliafito. Using google cloud vision in assistive technology scenarios. *IEEE Workshop on ICT solutions for eHealth 2016*, 2016.
- [3] Suhas Dwarakanath. <https://github.com/suhasd/thesis>.
- [4] Google Firebase. <https://firebase.google.com/docs/>.
- [5] Xiangqian Fu. Mobile phone ui design principles in the design of human-machine interaction design. In *2010 IEEE 11th International Conference on Computer-Aided Industrial Design Conceptual Design 1*, volume 1, pages 697–701, Nov 2010.
- [6] Anja Schmidt Gudrun Gorlitz. Digital signage: Informal learning in animal parks and zoos. *World Conference on E-Learning in Corporate Government Healthcare and Higher Education 2008*, pp. 841-846:841–846, 2008.
- [7] Z. He, B. Cui, W. Zhou, and S. Yokoi. A proposal of interaction system between visitor and collection in museum hall by ibeacon. In *2015 10th International Conference on Computer Science Education (ICCSE)*, pages 427–430, July 2015.
- [8] Y. Jeng, M. Hong, and H. Lu. The development of the mobile situated learning application based on microlocation technology. In *2017 6th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)*, pages 715–718, July 2017.
- [9] X. Li, D. Xu, X. Wang, and R. Muhammad. Design and implementation of indoor positioning system based on ibeacon. In *2016 International Conference on Audio, Language and Image Processing (ICALIP)*, pages 126–130, July 2016.

- [10] Donald A. Norman. Emotional design. *House of Electronics Industry*, pages 44–79, 2005.
- [11] T. Ogi, K. Ito, and S. Konita. Multilingual digital signage using ibeacon communication. In *2016 19th International Conference on Network-Based Information Systems (NBiS)*, pages 387–392, Sept 2016.
- [12] S. Rajkiran and C. Balakrishnan. Rssi estimation using bluetooth in smart phones. In *2015 2nd International Conference on Electronics and Communication Systems (ICECS)*, pages 1431–1434, Feb 2015.
- [13] P. Salunkhe, S. Bhaskaran, J. Amudha, and D. Gupta. Recognition of multilingual text from signage boards. In *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 977–982, Sept 2017.
- [14] M. Seraj and Chui Yin Wong. A study of user interface design principles and requirements for developing a mobile learning prototype. In *2012 International Conference on Computer Information Science (ICCIS)*, volume 2, pages 1014–1019, June 2012.
- [15] S. Sonth and J. S. Kallimani. Ocr based facilitator for the visually challenged. In *2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEEC-COT)*, pages 1–7, Dec 2017.
- [16] C. Takahashi and K. Kondo. Indoor positioning method for augmented audio reality navigation systems using ibeacons. In *2015 IEEE 4th Global Conference on Consumer Electronics (GCCE)*, pages 451–452, Oct 2015.
- [17] C. Wang, W. Su, and Y. Guo. An augmented reality mobile navigation system supporting ibeacon assisted location-aware service. In *2016 International Conference on Applied System Innovation (ICASI)*, pages 1–4, May 2016.
- [18] Wikipedia. Representational state transfer (rest) - <https://en.wikipedia.org/wiki/>.

- [19] Yang Yang, Zhouchi Li, and K. Pahlavan. Using ibeacon for intelligent in-room presence detection. In *2016 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA)*, pages 187–191, March 2016.