LOX Framework: Designing Human Computation Games to Update Street Views

Jongin Lee[1], John Kim[1,2], and Kwan Hong Lee[3,4]

[1] KAIST, Div. of Web Science and Technology, Korea
[2] KAIST, Dept. of Computer Science, Korea
[3] Redstar Ventures, USA
[4] POSTECH, Korea
{jongin.lee,jjk12}@kaist.ac.kr, kwantify@postech.edu

Abstract. Although the Web has abundant information, it does not necessarily contain the latest, most recently updated information. In particular, interactive map websites and the accompanying street view applications often have outdated information because street views change constantly and are very costly to update. In this work, we propose the LOX (Labeling and O/X) framework – a scalable human computation framework that discovers the latest updates of street views through mobile games. The challenge in this work is providing an interface to identify the differences between the latest street view and the outdated street view images. The LOX framework addresses this problem through three mobile games – a compatibility ESP (cESP) Game, a Drag&Drop Labeling Game and an O/X Game. The cESP and Labeling Games encourage users to identify new updates on outdated street view images while the O/X Game increases the precision of the differences identified by the users between the existing street view images and the latest street view. We conducted several user studies to assess the design and usability of the game. The collected data allowed us to assess the performance of utilizing human computation games to update new information while providing us guidance on how to design user interfaces and workflows to increase the quality of the data. The games provide an opportunity for significant cost savings to the service providers by providing an inexpensive method to determine which street views need to be updated.

Keywords: crowdsourcing, human computation, game with a purpose.

1 Introduction

Information that is available on the Web is constantly changing and, in many cases, does not accurately reflect the most current information in the real world. Mapping sites such as Google Maps have such limitations. In addition to simply providing street information and directions, the mapping sites provide different points of interests (POIs), restaurants, stores, reviews and street views. However, the information constantly changes (e.g. a restaurant might close and be replaced with a different store) and keeping the street view up-to-date by an organization

G. Memmi and U. Blanke (Eds.): MobiCASE 2013, LNICST 130, pp. 233–251, 2014.

can be extremely costly. Therefore, a grass roots manner that involves crowd sourcing is a very attractive approach to find updates and maintain the latest real world information.

The street views provide a panoramic view of a particular location from the perspective of someone walking along the street. It provides a close-up view of the area so that users can more easily identify their destination or discover the appearance of an area. Capturing and updating street view images require a vehicle equipped with specialized omniview cameras to drive around the streets. Based on the images collected, a panoramic view of the street is created. Since this is time consuming and costly[1], it is not realistic to always have the most recently updated view of the particular location. For example, in Google Street View, some street views from London are very recent as they are from 2012, but other major cities in the world are much older. The images from Paris were captured in 2008, Seoul in 2009, and New York City and San Francisco in 2011.

One Internet service provider in Korea (Daum) has a very similar service (called "Road View") and they provide information on the date that the current and prior road view images were obtained[1]. Based on this information, it can be known that some views near major cities are updated approximately every six months while other locations are updated every year or two. An example of how a street view changed over a period of two years is shown in Figure 1 for a major city within Korea. It is clear that some information (e.g. TGI Fridays) remains the same while other parts of the image (highlighted in the yellow square) have consistently changed during this time.

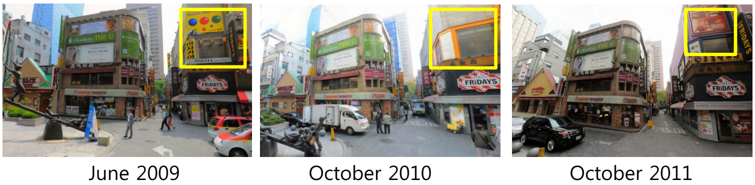

June 2009 October 2010 October 2011

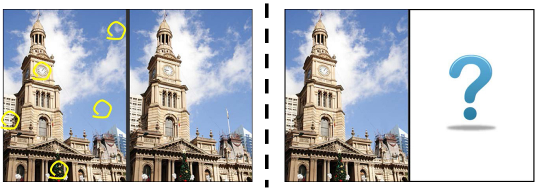
Fig. 1. An example of how a street view has changed over time

Because of this limitation of maps and street views, Google provides a Map Maker service[2] that allows individuals to submit changes to the current map information, including identifying new streets, new stores, new restaurants, etc. Once update requests are submitted, it requires another "expert" to verify and validate the request. However, this service is very labor-intensive and the process is not scalable and increases the difficulty in providing numerous updates. On the OpenStreetMap[3], map information is allowed to be modified by lots of contributors and seems to effectively maintain the latest map data. However, this service is limited to the traditional top-down view of the maps and does not

[1] It is estimated that Google spends $1-2 billion per year on street view to take photos all over the world. However, the cost will be much higher if Google Street View attempts to provide higher coverage as many cities outside of the US do not have street views available.

include the street views. With the wide spread use of smart phones and tablets, people are spending more time on their mobile devices and using them to navigate and find destinations. The value of grassroots community contribution is demonstrated by recent acquisition of Waze[4] by Google. Waze mobile application has shown great utility of community edited routes and maps – particularly to deal with traffic, constructions, potholes, and police radars. In our work, we enhance such capability by helping map service providers find updates or "differences" in the street views more easily through human computation games on mobile devices.

In this work, we attempt to identify the outdated information in the street view through a human computation game framework called LOX (**L**abeling Game and **O**/**X** Game) that is designed for users to identify differences between a stored image from existing street view database and the corresponding view in the current real world. The game is based on similar principles to the "Hidden Catch" game, which presents two images where the user has to identify their subtle differences (Figure 2(a)). This type of game is very straightforward and depending on the subtlety of the differences, the difficulty of the game increases or decreases. If necessary, the computer can easily solve this problem by comparing the images. However, if one of the reference images is not available (e.g. Figure 2(b)), then the Hidden Catch game cannot be completed by a computer. In this work, the LOX framework was inspired by similar principles to those in the Hidden Catch game, i.e. leveraging the user's familiarity of the street views in order to identify the differences and to obtain the updated information. In order to achieve this, the LOX framework consists of multiple games: an ESP game, a "labeling" game and a simple O/X game.



(a) (b)

Fig. 2. An example of (a) the Hidden Catch game and (b) our LOX game with an unknown image

2 Related Work

LOX Game follows the principles of game with a purpose (GWAP) [5] to facilitate finding of updates in street views of the physical world using a mobile game. Visual recognition and labeling has been a prominent activity for many human-computation researches. By providing the physical context to inform and fix the outdated online data is novel due to the wide availability of smart phones and street views at your fingertips. This addresses issues on what kind of scale and performance are needed for such human computation to be effective in reducing the effort to promptly collect data on these updates. There have been many research and commercial efforts on crowdsourcing through mobile phones. TxtEagle, deployed in Kenya, used SMS text messages to provide tasks like audio transcription, local language translation and market research[6]. MobileWorks provided OCR tasks that people in developing countries can play to earn wages anytime, anywhere. These applications show how these tasks can be accomplished on the mobile phone in a user friendly manner while contributing to the daily wages of people at the bottom of the pyramid[7].

As mobile phones have become more powerful computing and networked devices with sophisticated sensors and rich user interfaces, tracking peoples mobility behaviors, mapping out activity of urban cities, providing reviews, recommendations, and real time feedback on events happening in geographical locations have become mainstream[8][9][10]. Mobile phones have allowed people to participate in environmental participatory sensing such as spotting birds and their migration patterns or plants and their budding seasons[11], and bus arrival times that can be changed or updated in real time[12]. The projects have shown how widely deployed mobile phones and peoples contributions can provide the latest data in a very distributed manner and make the data collection more efficient than any centralized manner. LOX game attempts to engage general public through a mobile local game, utilizing their memory or physical presence at particular locations to capture latest changes on the street view data. Such labeling would allow street view creators to more efficiently gather new locations that need updated images.

3 Obtaining the “Unknown” Image

Following the principle of the Hidden Catch game, the biggest challenge in our LOX framework is to obtain the “unknown” image that corresponds to the latest street view image to identify the differences. With the growth of mobile devices with cameras, there are many street images available on the web and Google uses these images to shows streams of pictures when a particular location is searched on their map service. However, despite the availability of many of these images, it is very challenging to create a framework to discover the latest updates on POI’s based on these images because these pictures can be taken on different dates, resolutions, directions and angles. In addition, comparing street view images with individual pictures taken from different perspectives make the comparison a huge computational challenge making it not trivial to find images that contain any recently updated information.

Thus, to identify the updated spots in the “unknown” image for the game, there are two different possible sources for the unknown image.

1. Physical Image : The most ideal image is the “physical” image – i.e., if the user is physically present at the location of the street view, the unknown image is provided in front of the users and the street view (from the service

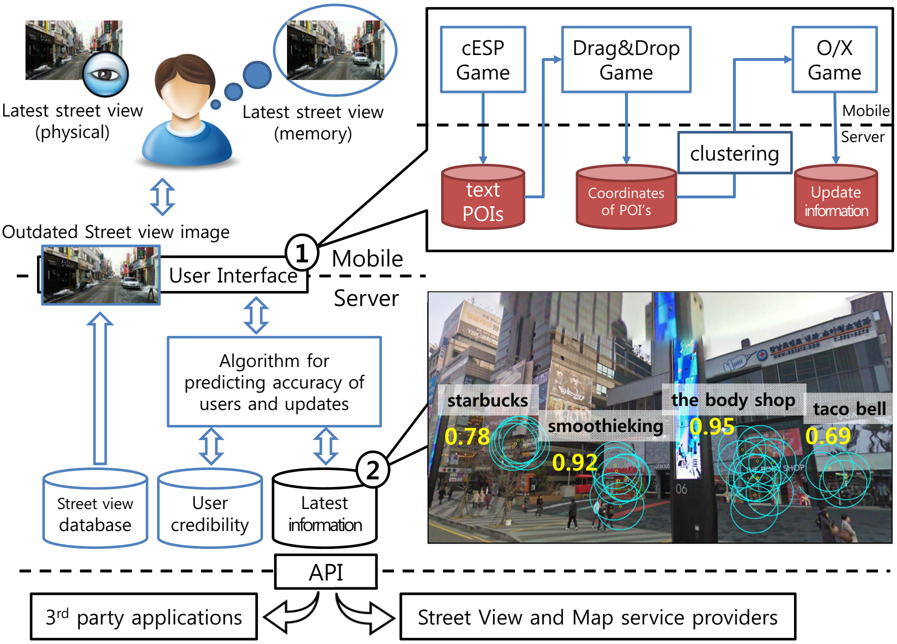


Fig. 3. High-level block diagram of the LOX framework

provider) can be directly compared against the physical image. Since we intend the game to be played on mobile devices, the game can leverage the GPS location information. For example, if a user is waiting for a friend at an intersection in a downtown district, the user can play the game to help discover latest updates needed in the street view images.

2. Memory Image : People are often very familiar with different street views – especially in areas around work, school, home, hangout places and perhaps places on their commute route [13]. Based on the user's familiarity with a given region, the "memory image" can be leveraged to play the LOX game and help identify the differences.

In this work, we focus on using the "physical" image within the LOX framework and we leave using the memory image to future work.

4 LOX Framework

A high-level block diagram of the LOX framework is shown in Figure 3. The goal of the LOX framework is not only to obtain the latest information but also to identify the *differences* with the existing street view images. The LOX framework consists of three games - the compatibility ESP (cESP) game, the Drag&Drop Labeling game and the O/X game. The purpose of the cESP game is to obtain text of new POIs for a given location. The Drag&Drop game uses information from Google Places to get coordinates of existing POI's. It also uses the text collected from the cESP game to augment with potential new POI's since Google Places does not necessarily have the latest, updated POI's. The information collected from these two games provides the potential latest POI information and where they might be located. However, we still need to identify whether certain parts of street view image are outdated. To provide the *differences* (difference between existing street view image and the latest actual street image), we introduced the O/X game.

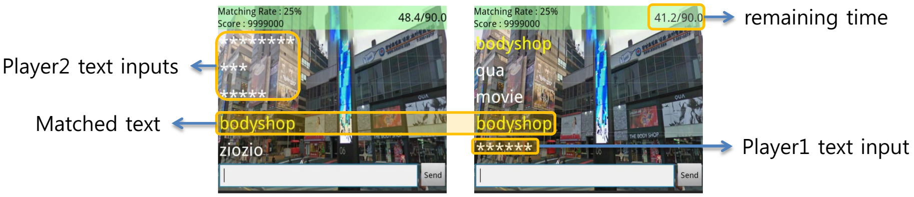
The high level description of the LOX framework is described in Figure 3. Since the three games are independent of each other, the games do not necessarily have to be played by the same player. For quality control, cESP game uses 'output agreement[14]' while both the Drag&Drop game and the O/X game compares the data to the data in the database, which enable players to play our game individually. 'Database agreement' is introduced to measure the credibility of the users and accept the contributions of the users in proportion to their credibility scores.

Table 1. Comparison of different games within the LOX framework

	cESP Game	Drag&Drop Game	O/X Game
Players	2 players	1 player	1 player
Summary	POI text collection	POI location identification	image processing
Genre	compatibility game	driving game	speed game
Quality control	output agreement	database agreement	database agreement
Input	street view images	street view images, POI's	street view images, locations of POI's, POI's
Output	latest POI's (text)	locations of POI's	differences between outdated and latest

4.1 Compatibility ESP (cESP) Game

The compatibility ESP (cESP) Game is a mobile version of the ESP Game [14] and the screen snapshot of cESP Game is shown in Figure 4. The original ESP game was designed to leverage human computation to label images by pairing up two random players and collecting labels that describe a given image. In our cESP game, the goals are still the same – have two players enter the same text to obtain "labels" for the features in the image. However, the labels are not describing the image (as in the original ESP game) but the collected labels from the cESP game are names and categories that identify the different POIs (e.g., Starbucks, T.G.I. Friday, etc.). In addition, a key difference with the ESP game is that instead of having two random players, we have two players that know each other play the game at the same time and leverage the simple "compatibility game" principle. The compatibility game [15], which is offered as a board game



(a) Player 1 (b) Player 2

Fig. 4. An example of two-player compatibility ESP (cESP) Game

but also played on- or off-line, attempts to evaluate how much compatibility exists between two people by how similar their answers are to different questions. We use this entertainment aspect to motivate the users to play the cESP game.

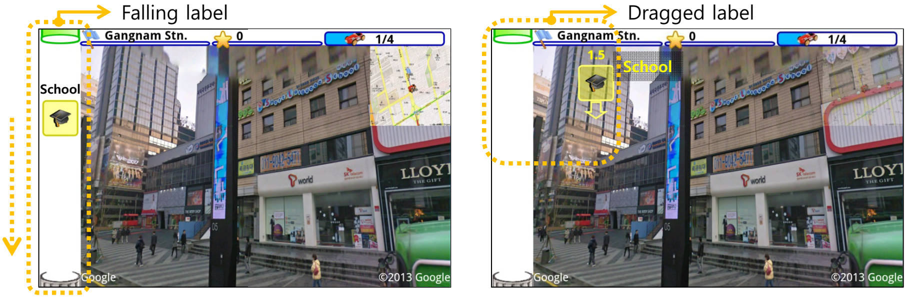
In our cESP, each player is shown a street view image and is assumed to be present nearby the street view physically. The two mobile players are connected to each other through bluetooth and each player enters any text that represent the POI's or signs one sees in the image. The labels typed by each player appears on the screen and the labels typed by the other player is also shown but as strings of "***". Once a label matches a label entered by the partner, the label is highlighted to signal "compatibility". The compatibility score depends on the number of matches and the sequential order of matches. Similar to the ESP games, we use the matched labels to update our database and they are used as inputs to the Drag&Drop Game.

4.2 Drag&Drop Game

The goal of the Drag&Drop game is to identify whether a label[2] exist or not in the latest or "current" street view and identify their locations on the street view images. The Drag&Drop game leverages 1) the intuitive concept of drag and drop which is a popular gesture in touch user interfaces [16](i.e., a given label should be dragged and dropped at the right coordinate to map the label based on the current world) and 2) the driving game. To provide similar experience as a driving game, the street view "moves" toward the destination and the route is shown on the map located at the upper right corner of the screen (Figure 5(a)). For a given street view image, labels drop down the left side of the screen and while they fall, the player has several options:

1. Select and drag: The user selects or grabs the falling label and drags it over the street view to identify within the limited time (Figure 5(b))
2. Touch and delete: If the user determines that the given label is not available in the given street view, the user can touch the falling label and it will be marked for deletion.

[2] Labels can be different categories of POIs (e.g., coffee store) as well as particular names of the POIs (e.g., Starbucks).



(a)

(b)

Fig. 5. The user interface of a falling a label(a) and a dragged label(b) in the Drag&Drop Game. The fingertip is touching the screen right below the arrow pointing down underneath the label in (b).

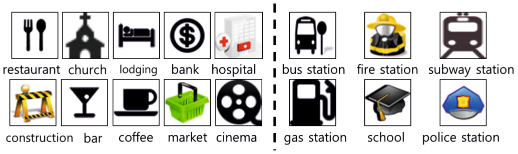


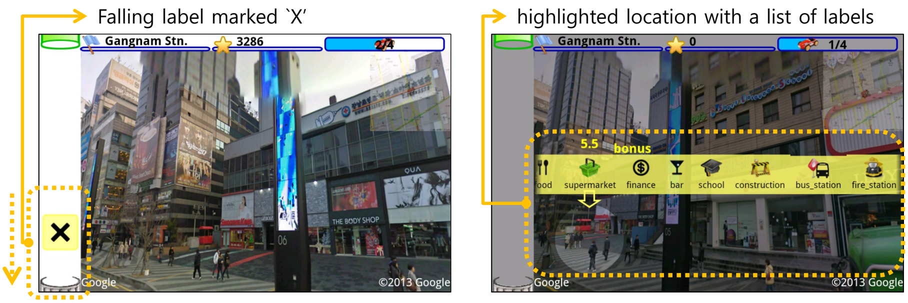
Fig. 6. Examples of different label categories used in the Drag&Drop game

3. Do nothing: If the user is uncertain about the label, the user can simply let the label fall through to the bottom.

The basic user interface is shown in Figure 5(a). At the top, the location name of the place of street view image, game score and the progress bar are shown. Top-down map with the current location and the destination is shown at the upper right corner. This map becomes transparent when a player touches any area on the screen. Two types of labels are introduced in the game: a category label (as shown in Figure 6) and a 'text' label where the 'text' is obtained from the cESP game.

Designing the User Interface. The user interface of the Drag&Drop Game evolved for better interaction with players based on feedbacks and observations from several early user studies. There were 5 significant updates from the first iteration of the user interface:

1. The concept of falling labels was introduced to enforce players to drag the labels to the right location in a limited time. Left column area for the falling labels replaced the remaining time bar which had occupied the top area of the screen and increased the available device screen area.
2. In the initial driving method street view image changed into the next image abruptly after finishing marking of given labels. Although several parts of the street view image overlapped and changed into the next street view



(a) (b)

Fig. 7. Falling label is marked as ‘X’ (a) when the label is touched and marked for deletion. For the bonus, when the label is dragged and dropped, several labels pop up like (b) and wait for user’s selection.

image, players pointed out that they could not feel moving forward and even indicated that the images did not seem to be related to each other. Thus, for better driving experience, animation was used to transition from one street view image to the next street view image.

3. The labels were designed with an arrow that points down to where the users are touching the screen. Otherwise the players cannot recognize the feature on the image because it becomes hidden by the label icon.
4. To give feedback on the coordinates specified by the players, a star icon was shown at the coordinate of the label after dropping the icon.
5. Because some players tend to drag a label into the hole on the left bottom of the device screen (i.e., they want to cancel or delete a label while dragging the label), later version of the game allowed this action.

Bonus Label. After an initial implementation of the game, one problem was the lack of “proper” labels – e.g., the user knows that a new convenience store exists at a given location but a convenience store label does not drop. This can be frustrating to the user but also problematic for the game itself since it prevents us from obtaining valuable information. To allow this (while keeping the game simple), we introduce a bonus label as shown in Figure 7. When this bonus label appears, the user can drag and drop it to any location where the user feels a new feature exists. After the bonus label is dragged and dropped, as (b) in Figure 7 a list of different labels pop up so that the user can select an appropriate label.

4.3 O/X Game

The information collected from the Drag&Drop game provides latest street view information but does not provide the differences between the latest street view image and the street view image from the service provider. The difference information can be very useful to the service providers – for example, the information can be useful

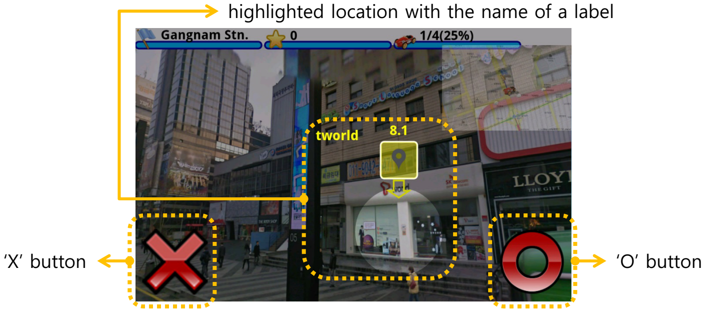


Fig. 8. User interface for the O/X Game

as a guidance to identify regions where the street views need to be re-captured. As a result, some form of image processing is necessary to identify the differences.

To provide the image processing, we leverage a simple O/X Game. The O/X game is based on the same principle as O/X quiz (or True/False quiz). In traditional O/X (or True/False quiz), the user is provided with a statement and is required to respond either with TRUE ('O') or FALSE('X'). The interface for the O/X Game is shown in Figure 8. Similar to the Drag&Drop game, the user is also shown a street view but instead of a falling label, the user is provided with a label and an arrow pointing to a given location. To focus on the region of interest, the image is shaded aside from the region in the circle and the arrow. At the bottom of the screen, there are two buttons – an 'O' button and an 'X' button. Based on the shown image, the user is asked to select the 'O' (if the label correctly identifies the feature of interest) or the 'X' otherwise. Thus, an 'X' output from the O/X Game is interpreted as a "difference" in the street view or POIs that have changed. Unlike the cESP game and the Drag&Drop game, which either requires familiarity with the street view or locally being present at the location, this game does not require any familiarity with the region. It works simply based on both the shown labels and the street view image to identify whether the label matches the shown feature on street view image.

5 Algorithm

There are two significant challenges in the LOX framework: 1) can the user selection be trusted? (i.e., user credibility) and 2) how accurate are the labels that are currently in the database? (i.e., label accuracy). These two scores are interrelated as an input from a more credible user will increase the label score while an input from a less credible user will not significantly impact the label accuracy score. In this section, we discuss how these two components are determined and their impact on overall performance of LOX framework.

5.1 User Credibility

The *user_credibility* has a value between 0 and 1 and shows the degree of a user's right prediction based on current database($label\_score_n$). The user who

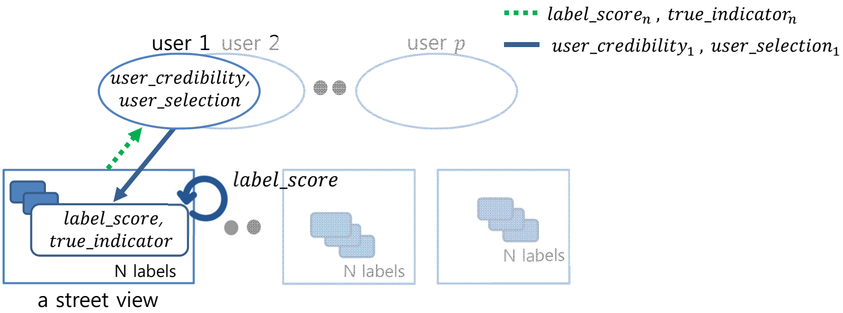


Fig. 9. Relationship between *user_credibility* and *label_score*

performs better on the prediction has a higher *user_credibility*. Given N labels in a street view image, a participant's *user_credibility* is measured with the dataset $\{(label\_score_n, s_n, t_n),\ n = 1, ..., N\}$ where $t_n \in \{1, 0\}, s_n \in \{1, 0\}$. s^n and t^n are shorthand for *user_selection* and *true_indicator* respectively. Each label has a confidence variable(*label_score*) and its corresponding tentative true indicator(t)(e.g.,if a *label_score* is higher than 0.5 then t is 1; otherwise, t is 0). User's selection(s) is decision between 'true' and 'false' about a label. The formula is described as :

$$user\_credibility = \frac{\sum_n \mathbb{I}[s_n = t_n] score(label\_score_n, t_n)}{\sum_n score(label\_score_n, t_n)} \tag{1}$$

$$score(label\_score, t) = \begin{cases} label\_score, & \text{if } t = 1 \\ 1 - label\_score, & \text{if } t = 0 \end{cases} \tag{2}$$

The function *score* is introduced to deal with both true and false *label_score* fairly. Each player is encouraged to predict each label's t_n by selecting s_n in the LOX games. Therefore, the more a player matches the pair of t_n and s_n, the higher *user_credibiltiy* the player gets. Green dotted line of Figure 9 shows how *user_credibility* flows in the over game.

5.2 Label Scores

The *label_score* has a value between 0 and 1, with a value closer to 0 representing a label that is likely to be false while a label with a value closer to 1 representing label that is likely to be true. The *label_score* changes based on the players' selection(s) and the player's *user_credibility*. Thus, every time *user_credibility* and user selection(s) are added, *label_score* increases or decreases by the amount represented by *user_credibility* (An example from the user study is shown later in Figure 12).

Given p users for a label, the *label_score* is the ratio of the sum of a group of *user_crediblity* whose selection is 1 to the sum of a group of *user_credibility* whose selection is 0 or 1 (i.e., How many players regard the label as 1). Then, the

$label\_score$ is measured with the dataset $\{(user\_credibility_n, s_n), n = 1, ..., p\}$ and the formula is described as :

$$label\_score = \frac{\sum_n \mathbb{I}[s_n = 1] user\_credibility_n}{\sum_n user\_credibility_n} \tag{3}$$

Two blue lines of Figure 9 show how *label_score* flows with a set of *user_crediblity* and s. After reallocating *label_score*, true indicator(t) is also updated according to the *label_score* (e.g.,if the *label_score* increases above 0.5, t is modified as 1; otherwise, t is 0).

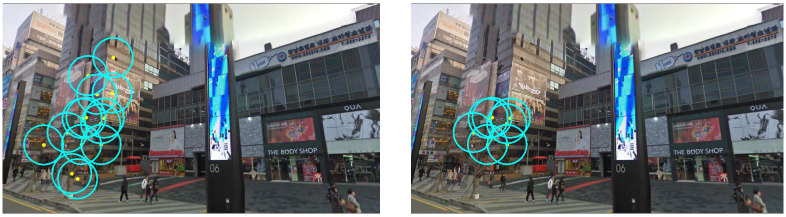
Initializing Label Scores. The *label_score* of the Drag&Drop and the O/X game need to be initialized. In the Drag&Drop game, the score of category label (as shown in Figure 6) is relatively determined by the frequency of the extracted information from Google Places API (e.g., extracted multiple category of POIs can be used as strong signal of existence of the 'category'). The POIs which do not change frequently can be used strong signal of existence (e.g., right part in Figure 6). Another type of label in Drag&Drop game is *text* label which are the output from the cESP game. The *label_score* of the type is relatively initialized based on the frequency of the matched numbers. In the O/X game, the score of labels from the Drag&Drop game is initialized as '0.5'. as it is difficult to determine if a label (or POI) has changed or not. As a result, we also introduce *random* labels whose location is randomly selected. The random labels' *label_score* is initialized as a low value (0.3) to represent it is very likely to be a wrong label.

5.3 Identifying Location and Number of POI's

A simple clusting algorithm is needed to identifying location and the number of POI's captured. Common k-means algorithm could not be used since it needs the parameter k known ahead of time. Instead, DBSCAN [17] algorithm was used because of two reasons : 1) it does not need to specify the number of clusters and 2) some noises are not selected during the clustering. Because the algorithm is density-based clustering, two parameters are required : ε (eps) and the minimum number of points required to from a cluster(minPts). ε (eps) can be easily decided by the size of touched finger point and the minPts is configured as '1'(i.e., we assume that some positions where at least two people agreed on are accepted). Figure 10(a) shows the raw coordinates from players and Figure 10(b) is their clustered result by DBSCAN. Although the cinema account for the several floors of the building, the final result shows the centroid of the building.

5.4 Cheating and Filtering Noise

In any human computation game, cheating needs to be addressed. In this section, we describe different methods in LOX framework that help identify such behavior.



(a) raw coordinates of labels (b) clustered coordinates by DBSCAN

Fig. 10. Example of result of clustered positions by DBSCAN

cESP Game. Players of the cESP Game can input text that is not related to street view. However, the output of cESP is not the final results but input to the Drag&Drop gaem and thus, this process can help filter out some of the noise in the input. Global count of each word is maintained to prevent group of players from typing same word. In addition, some players might type the same texts cooperatively but unlike the original ESP game, this does not necessarily degrade quality of the text collected.

Drag&Drop Game. Players of the Drag&Drop game can drag or delete labels randomly. To prevent this, *user_credibility* is used to measure the credibility or accuracy of the player's selection based on the current database. The input or selection from players with low *user_credibitliy* will have small impact on the database. One observation that we made from the user study was that players who have higher *user_credibility* took relatively short time (i.e., milliseconds) in determining 'X' or 'drag and drop' – most likely because the player was confident in the decision. This information can likely be incorporated into the user's credibility.

O/X Game. Players of the O/X Game can select 'O' or 'X' randomly. To minimize this impact on the database, we introduce random false labels – i.e., labels with very high probability that it should be 'X'. In addition, for some labels, we introduce the same label twice within the same game to determine if there is an agreement in the player's response. These approach helps to properly calculate the credibility of the player.

6 User Study

To evaluate the feasibility and performance of LOX framework, we performed a user study near Gangnam subway station[3] in Korea for 5 days. To access the "real" street view, participants were randomly recruited at two coffee shops that had windows facing the streets. Participants played one of the LOX games and

[3] We choose Gangnam subway station because it is one of the most popular places where people hang out during the day in Korea.

then filled out survey for feedback and demographic information. The purpose of the game was not explained to the participants. Total of 4 street view images dated 2009 were obtained from the Google Street View image API and used in the study. Android mobile device (Samsung Galaxy Note) was used for the user study. 62 participants participated in the user study (Table 2) – with 42 participants for the cESP and Drag&Drop game in the Gangnam area and 20 participants who were not familiar with the given street view images played the O/X game.

Table 2. Demographic information of participants within the LOX framework

	cESP Game	Drag&Drop Game	O/X Game
# of Participants	20	22	20
Woman(%)	11(55%)	11(50%)	1(5%)
Man(%)	9(45%)	11(50%)	19(95%)
Age group($20 < age \leq 30$)	19	22	20
Age group($30 < age$)	1	0	0

6.1 LOX Framework User Study Results

The result of the cESP game is shown in Figure 11 and shows the number of typed words from each player, and the number of words matched. The matched texts of POI were used as input to the Drag&Drop game.

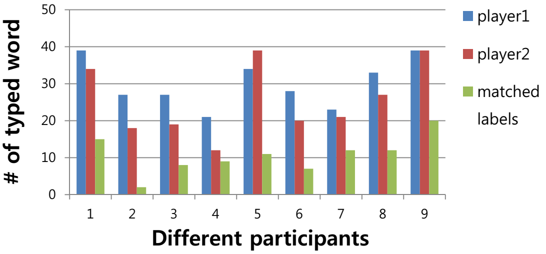
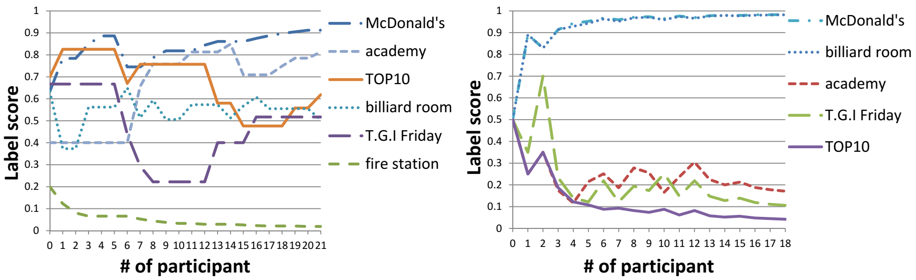


Fig. 11. Different matched numbers in the cESP game

From the Drag&Drop game, 132 labels from four street view images were collected. 56 labels whose score was above 0.6 are assumed to be true and the clustered labels are passed to the O/X game. Results from Drag&Drop game are shown in Figure 12(a). Labels which did not change such as ‘McDonald’s’ – which correspond to labels that were available in the street view database – converged very quickly to a value of 1. In addition, some of the other obvious labels (‘fire station’), which did not exist in the street view database image, converged quickly to a value of 0. As for the other labels, the scores fluctuated.

Some results of O/X game are shown in Figure 12(b). Scores of the different labels mostly converge as more participants (x-axis) played the game. The results



(a) (b)

Fig. 12. Score of different labels and how they evolved in (a) Drag&Drop Game and (b) O/X Game (TOP10 is clothing store.)

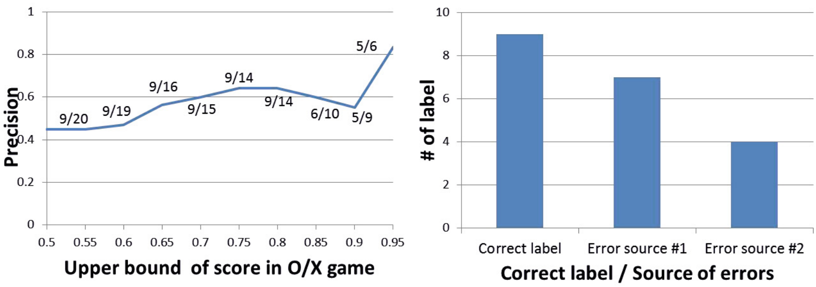
show that for two labels, the scores converges to score of 1 which identifies that they are ‘O’ or labels that have not changed and shown on current street view database image. For the other three labels, they converge to a score closer to 0 signify that they are ‘X’ or “differences” and information that is not available from the service provider street view database image.

6.2 Result Validation

We evaluated the accuracy of the labels that were collected by manually verifying the new POIs. An example comparing the labels obtained from LOX with a manual picture taken from the same location is shown in Figure 14. We focus the analysis on the 20 labels that are the outputs of O/X game with a score higher than 0.5. The precision shows correctly identified new labels – i.e., for example, 9/19 precision refers to 19 labels were identified through the LOX framework but 9 were manually verified to be correct new POIs (i.e., 10 labels were incorrectly identified.) The score reflects the confidence of the new labels that we obtain. As the results show in Figure 13(a), if we only consider labels with very high scores (i.e., scores higher than 0.95), the accuracy is over 80% but if we consider all labels with score of higher than 0.7, the accuracy drops to approximately 60%. Thus, by maintaining a high threshold, the accuracy of the label obtained from LOX can be much more accurate.

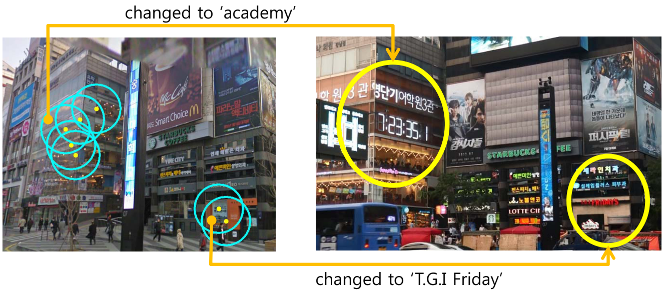
To understand the “errors” or the incorrect labels, we analyzed the false labels and classified the source of errors into one of the following three categories(Figure 13(b)), based on the data collected.

1. Since the street view image is often captured from a car, some obstacles (e.g. street light, trees, or people) can hide some part of the street view image and make it difficult to identify some POIs. This error type mostly occurred in the O/X Game.
2. Another source of error was difficulty in identifying the exact location of the POIs. For example, a theatre was located in a building, occupying several



(a) (b)

Fig. 13. The final result according to the upper bound of score(a) and the types of the final results of labels(b)


changed to 'academy'

changed to 'T.G.I Friday'

Fig. 14. Example of new labels(English academy and T.G.I Friday) in the game(left) and their corresponding real locations in the real street view(right)

floors but some participants were confused where to drag the labels to. This error type occurred mostly in the Drag&Drop game.

3. After manual analysis, we found the off-line cluster algorithm also caused some errors (but the number of errors was relatively small, compared to the other two error types.)

6.3 Survey

We did a post user-study survey to capture the demographic information, their experience with given street view images, experience with mobile games, evaluation of how fun or was user interface, and some feedback on games. Almost all participants had experience playing mobile games – more than 80% of participants played mobile games twice a month.

The survey on the user interface and entertainment of the different games within the LOX framework are shown in Figure 15. Although the purpose of the game was not explained, approximately 30% of the participants agreed that they want to play the game again for entertainment purpose. Overall, all the games were shown to have easy user interface. It was interesting to note that the O/X game had the simplest user interface but not necessarily the most interesting game, according to the survey.

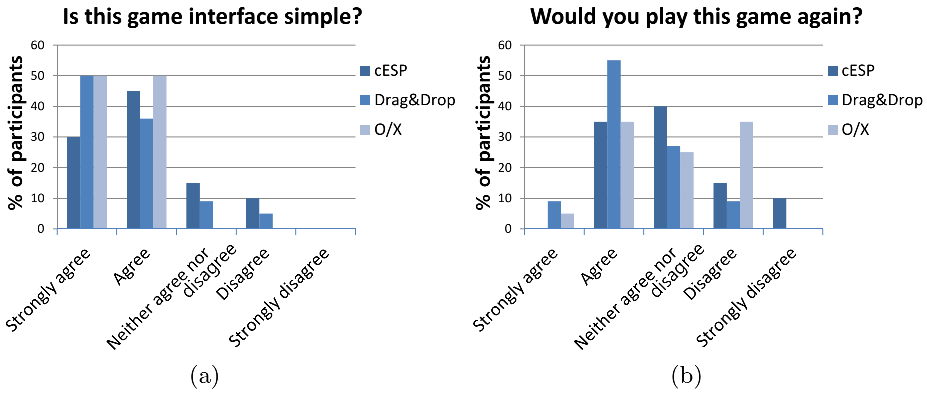


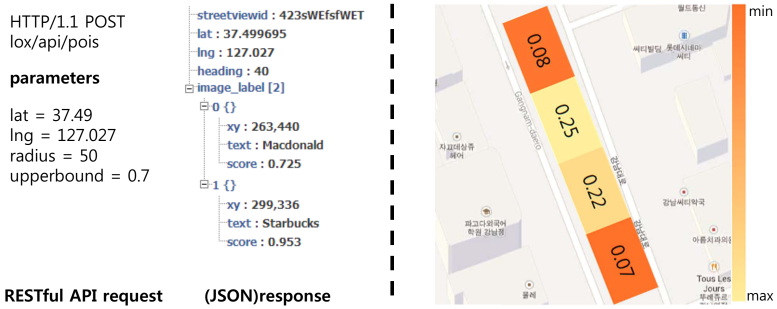
Fig. 15. Survey results on (a) LOX user interface and (b) entertainment aspect of the game

Feedback. In terms of additional feedback, some of the participants found the games interesting since it involved "real" street views, compared with most other mobile games. The two player game approach also appealed to some participants. In addition, perhaps a significant contribution of this game came from a participant who commented "I become more aware of the street POIs after playing the game." This initial interest in game can provide additional motivation to play the game again in the future.

6.4 Potential Applications

In this section, we describe how the outputs of LOX framework (either the output of the Drag&Drop game or the output of the O/X game) can be used. The analyzed output of LOX can be provided through an API for use by 3rd party application and an example is shown in Figure 16(a). RESTful API is a web API design model that is relatively simple and scalable [18] and we provide a RESTful API so that applications and service providers can make use of LOX data. Figure 16(a) example shows how the output of LOX framework can be presented – which include providing the information and location of the latest POIs. For each POI, the "score" determines the confidence of the new POI. And other API interface can provide additional street view information.

The output from LOX can also be leveraged by map service providers to identify how much the street view has changed. An example of such output is shown in Figure 16(b) where the higher intensity identifies regions with larger amount of changes or *differences*. The amount of change is calculated as a fraction of the POIs that have changed. The service providers can use this information to determine which regions need to have the street views updated. In addition, we expect the outputs of LOX can be leveraged for other purpose. For example, demographic information obtained from LOX games can be used for marketing purposes and while the latest city information can also be leveraged by city or urban planners.



```
HTTP/1.1 POST
lox/api/pois

parameters

lat = 37.49
lng = 127.027
radius = 50
upperbound = 0.7
```

RESTful API request

```
streetviewid : 423sWEfsfWET
lat : 37.499695
lng : 127.027
heading : 40
image_label [2]
  0 {}
    xy : 263,440
    text : Macdonald
    score : 0.725
  1 {}
    xy : 299,336
    text : Starbucks
    score : 0.953
```

(JSON)response

(a)

(b)

Fig. 16. (a) Request and response from REST API and (b) the heatmap showing the amount of changes

7 Conclusion

In this work, we proposed the LOX framework that consists of human computation games to provide updates to point of interests (POIs) on the street view maps. The LOX framework consists of different games that are used to help provide the text for the latest POIs, identify their locations on the street view maps, and finally help identify how much they differ from the existing street view maps. Based on the user studies, we were able to collect new information on the latest POIs while identifying the amount of differences with the current street view images provided by the service providers. Although the main purpose of the LOX framework was to provide updated information to the often outdated street view information, there are many other applications that can make use of the output of the LOX framework – with applications ranging from marketing to city/urban planning. We plan on exploring these opportunities as part of future work.

Acknowledgement. This research was supported in part by the SW Computing R&D Program of KEIT (2013, No.10041313, UX-oriented Mobile SW Platform).

References

1. Daum road view, `http://map.daum.net/`
2. Google map maker (2013), `http://www.google.com/mapmaker`
3. Openstreetmap, `http://www.openstreetmap.org/`
4. Waze mobile application, `http://www.waze.com`
5. von Ahn, L., Dabbish, L.: Designing games with a purpose. Commun. ACM 51(8), 58–67 (2008)
6. Eagle, N.: txteagle: Mobile crowdsourcing. In: Aykin, N. (ed.) IDGD 2009. LNCS, vol. 5623, pp. 447–456. Springer, Heidelberg (2009)

7. Narula, P., Gutheim, P., Rolnitzky, D., Kulkarni, A., Hartmann, B.: Mobileworks: A mobile crowdsourcing platform for workers at the bottom of the pyramid. In: AAAI 2011 Workshops, vol. WS-11-11 (2011)
8. Kanhere, S.S.: Participatory sensing: Crowdsourcing data from mobile smartphones in urban spaces. In: Hota, C., Srimani, P.K. (eds.) ICDCIT 2013. LNCS, vol. 7753, pp. 19–26. Springer, Heidelberg (2013)
9. Lane, N., Miluzzo, E., Lu, H., Peebles, D., Choudhury, T., Campbell, A.: A survey of mobile phone sensing. IEEE Communications Magazine 48(9), 140–150 (2010)
10. Celino, I., Cerizza, D., Contessa, S., Corubolo, M., Dell'Aglio, D., Valle, E., Fumeo, S.: A social and location-based game with a purpose to crowdsource your urban data. In: PASSAT 2012, pp. 910–913 (2012)
11. Budburst mobile project, `http://networkednaturalist.org/budburstmobile/`
12. Zimmerman, J., Tomasic, A., Garrod, C., Yoo, D., Hiruncharoenvate, C., Aziz, R., Thiruvengadam, N.R., Huang, Y., Steinfeld, A.: Field trial of tiramisu: crowdsourcing bus arrival times to spur co-design. In: CHI 2011, pp. 1677–1686 (2011)
13. Quercia, D., Pesce, J.P., Almeida, V., Crowcroft, J.: Psychological maps 2.0: a web engagement enterprise starting in london. In: WWW 2013, pp. 1065–1076 (2013)
14. von Ahn, L., Dabbish, L.: Labeling images with a computer game. In: CHI 2004, pp. 319–326 (2004)
15. Lawrence, E.C.: Compatibility game, [US Patent 5,681,046] (October 28, 1997)
16. Shneiderman, B.: Designing for fun: how can we design user interfaces to be more fun? Interactions 11(5), 48–50 (2004)
17. Sander, J., Ester, M., Kriegel, H.-P., Xu, X.: Density-based clustering in spatial databases. Data Mining and Knowledge Discovery 2(2), 169–194 (1998)
18. Böck, H.: Restful web services. In: The Definitive Guide to NetBeans^{TM} Platform 7, pp. 345–352 (2011)