A Proximity Information Propagation Mechanism Using Bluetooth Beacons for Grouping Devices

Masato Watanabe, Yuya Sakaguchi, Tadachika Ozono, Toramatsu Shintani
Department of Scientific and Engineering Simulation
Graduate School of Engineering, Nagoya Institute of Technology Japan
masatow@toralab.org
syuya, ozono, tora@toralab.org

ABSTRACT: *In educational or business environments, we often have a paperless meeting by using tablet devices. The meeting participants need to log into a meeting server of the meeting by using a token, a keyword to log into a meeting, defined for each meeting. Nevertheless, the meeting can be immediately held if the tablets ready for use by reducing the costs of logging in of each participant. To reduce the costs, we proposed a new grouping method of tablet devices. The grouping method uses proximity information from a Bluetooth beacon to recognize grouped devices that are near each other. Moreover, the method can solve a problem that each device can be an obstacle attenuating signals for grouping. By using the method, working efficiency with tablet devices can be improved because the attendees do not need to operate each device individually to group them. Therefore, the devices can be grouped just by gathering them in a specific nearby location. We observed experimentally that the mechanism can resolve problems of reduced the attenuation of Bluetooth signals received by all immediate beacon receivers.*

Keywords: Bluetooth, Proximity information propagation, Device grouping, Tablet device

Received: 4 September 2016, Revised 9 October 2016, Accepted 16 October 2016

1. Introduction

In educational or business environments, we often have a paperless meeting by using tablet devices. The meeting participants need to log into a meeting server of the meeting by using a token that is defined for each meeting. The token is a keyword to log into a meeting. The meeting can be immediately held if the tablets ready for use by reducing the costs of logging in of each participant. To reduce the costs, we propose a new grouping method of tablet devices. By using the method working efficiency with tablet devices can be improved. The grouping method uses proximity information from a Bluetooth beacon to recognize grouped devices that are near each other. The attendees do not need to operate each device individually to group them. The

devices can be grouped just by gathering them in a specific nearby location. When using proximity information based on Bluetooth in grouping methods, radio attenuation can hamper the operation if a device is set between the receiver and the transmitter. For example, if a cluster of devices attenuates the signal because they are situated too close to each other, the devices cannot be grouped as a cluster. It can also be difficult to distinguish one cluster from another cluster. That is, if two clusters of devices are both set at two nearby points, it can be difficult to decide if there is one cluster or two clusters.

In this study, we present a method to solve these two problems with grouping methods using proximity information based on Bluetooth. To solve the first problem, we developed a proximity information propagation (PIP) mechanism. Using the PIP mechanism, a device can communicate sequentially with each of the other devices in a cluster. To verify whether the second problem was solved, we measured the received signal strength indicator (RSSI) for each device in two clusters of devices. In Section 2, we discuss existing device grouping methods and their advantages and disadvantages. In Section 3, we discuss how to implement the PIP mechanism and its applications. In Section 4, we discuss an experiment to check the radio attenuation by other devices and a pilot study to distinguish nearby groups.

2. Existing Device Grouping Methods

2.1 Device Grouping using existing methods

We discuss three problems on grouping devices by using a Bluetooth beacon. The first problem is an uncontrollable geo-fence problem, the second problem is an undesirable radio attenuation problem, and the third problem is a group separation problem. Fig. 1 (a) depicts the uncontrollable geo-fence problem, and Fig. 1 (b) shows the undesirable radio attenuation problem and the group separation problem. These cases represent situations demonstrating the three problems we face when grouping devices.

The geo-fence is a virtual barrier to define geographical boundaries. One of the advantages of a Bluetooth beacon is that it composes a small, less than 10 m, geographical boundaries based on signal attenuation. The geo-fence is based on radio attenuation of Bluetooth signals. The radio attenuation is caused by distance or obstacles, and this attenuation prevents the grouping of devices.

As shown in Fig. 1 (a), there are five iPads; the iPad held by the user is a transmitter to make a geo-fence for grouping, and the others on the table are receivers. The receivers are arranged next to each other and receive radio signals, and the RSSI of the transmitter is measured. If the RSSI of a receiver is greater than a predefined threshold, the receiver can enter the geo-fence.

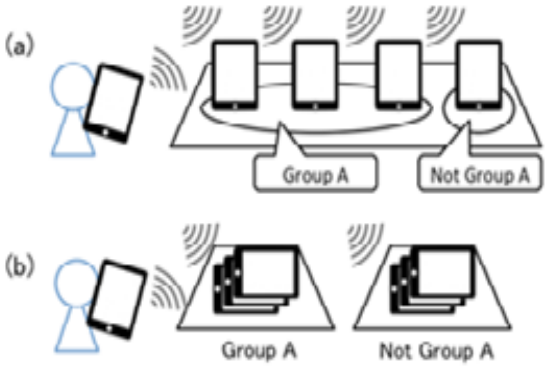


Figure 1. Grouping devices gathered in one place using Bluetooth beacons

The uncontrollable geo-fence problem means that users cannot recognize which devices are grouped. The user can make a small group of devices in the same geo-fence of a Bluetooth beacon, however, the user cannot accurately predict which devices are grouped. The reason is that the strength of a Bluetooth signal is invisible and variable.

In addition, it is difficult to distinguish between two groups in a situation where a transmitter is located the same distance from two clusters even if they are separated from each other. The undesirable radio attenuation problem is a problem that a receiver

can be an obstacle to make undesirable radio attenuation. For example, if receivers are stacked, RSSI can be unpredictable. It means that it is difficult to make a proper threshold to identify a group.

The undesirable radio attenuation problem causes the group separation problem. In another configuration, as shown in Fig. 1 (b), there are two stacks of iPads, beacon receivers. A beacon transmitter is held by the user. The user wants to make two groups for each stack. The receivers determine their groups by measuring the RSSI of the transmitter. Against someone's better instincts, the top device of each stack will be grouped because the top devices attenuate the beacons creating the geo-fence. Since the second and later devices from the top receive the weaker signals, they cannot enter the group.

2.2 Device Grouping using existing methods

To realize device grouping, the chosen mechanism must recognize each group, and the devices belonging to it. Here we discuss previous studies related to identifying devices. He's system[1] used iBeacon as geo-fence. A user with an iBeacon receiver can benefit from the system by approaching an i Beacon transmitter. In the situation shown in Fig. 1 (a) and (b), device grouping by using geo-fence is difficult, because we cannot correctly control a coverage of geo-fence, and the devices attenuate the beacons creating the geo-fence.

Kao [2] developed a small nail-form input interface. Input to the interface is sent to a personal computer or a smart-phone via a signal in the Bluetooth Low Energy (BLE) standard. Kao's system interface used BLE as its communication technology. We can apply our mechanism to a user interface following Kao's approach. Our mechanism can help extend the small user interface and implement more convenient functions.

Aumi[3] suggested a device-selecting-method using the Doppler effect to identify devices. To use the Doppler effect, the user device must emit a sound. A gesture, such as shaking the device, pushes the device towards another device that the user selects, activating the Doppler effect. When the device receiving the sound recognizes the Doppler effect, the device is selected. This method has broad applicability because all devices with a speaker and a microphone can use the method. In addition, the gesture enables one device to identify another device. In our study, we introduce a constraint in the distance between the devices when identifying a device.

Suzuki [4] suggested Pair Swipe, which identifies devices that interact with other device by swiping screens from one device to another. Pair Swipe has the advantage of an intuitive operation to identify devices. However, Pair Swipe cannot identify multiple devices at the same time. If we want to identify multiple devices, we need to swipe each devices that needs to be identified. With our mechanism, one device 's receiving beacon signal enables it to interact with another device. All devices receiving the beacon signal, interact with another without user interaction.

Goel [5] developed a system that achieves inter-device communication via natural gestures. For example, we could make two devices on a shared table communicate by tracing a line between the two devices with a finger. To detect the devices on a shared table, Goel's system uses touch sensors (an acceleration sensor and a vibration motor).However, inter-device communication should be possible without an additional sensor. With additional sensors, the larger the number of devices, the greater the cost to operate them. We lower the cost by using the transmission and receiving functions that Bluetooth already has. These functions are standard in common devices.

3. Proximity Information Propagation Mechanism Using Bluetooth Beacons

3.1 Base Technology

This mechanism uses a beacon information provided by a beacon using the 2.4 GHz band. After receiving beacon B, a beacon receiver decodes the beacon information I_B from beacon B. I_B has three types of ID: universally unique identifier (UUID) , Major, and Minor. The beacon receiver calculates the RSSI and a proximity state from the received beacon. The proximity state is an evaluation value of the distance between the beacon transmitter and the beacon receiver. The proximity state has four types of evaluation values: I*mmediate, Near, Far* and *Unknown*. The four proximity state are put in order of distance between the beacon transmitter and the beacon receiver, *Immediate, Near, Far* and *Unknown.* Putting them in order from the narrowest range, *Immediate, Near, Far* and *Unknown*.

Fig. 2 shows the architecture of the mechanism. This mechanism is implemented as an iOS application library and can be combined with another module O developed by an iOS application developer. The solid arrows in Fig. 2 indicate the mechanism's

workflow. The dashed arrows in Fig. 2 show the input-output in the case where the module cooperates with another module O.

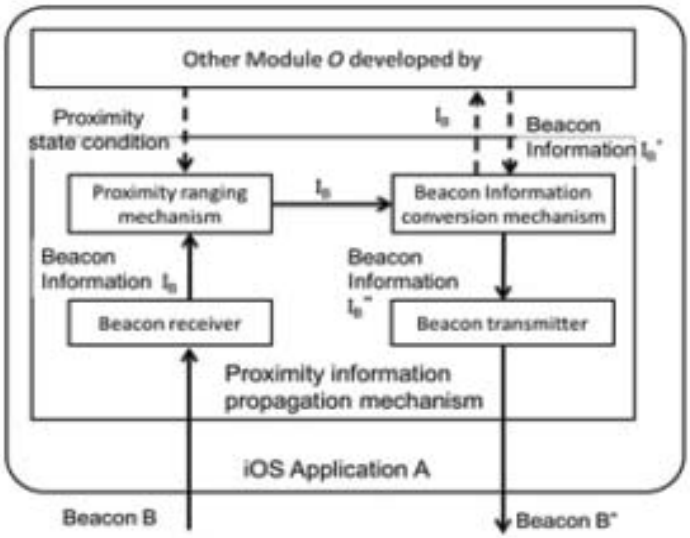


Figure 2. Proximity information propagation mechanism

When a beacon receiver component receives beacon B, it extracts the beacon information I_B. The receiver component inputs I_B to the proximity ranging mechanism. The proximity ranging mechanism estimates whether the proximity of I_B meets the proximity state condition. The proximity state condition is set to I_B. proximity ≤ *Immediate*, I_B. proximity ≤ *Near*, or I_B. proximity ≤ *Far*. I_B. proximity ≤ *Immediate* means that the proximity state of I_B is near than Immediate. The other module O can modify the proximity state condition. If and only if the proximity of I_B meets the proximity state condition, the receive component inputs I_B into the beacon information conversion mechanism. The beacon information conversion mechanism inputs I_B into another module O and receives I_B', which is calculated by O. The beacon information conversion mechanism assigns I_B' to I_B" and inputs I_B" into the beacon transmit component. The beacon transmit component generates a beacon B" and transmits B". We call these processes proximity information propagation.

When proximity information propagation occurs in the devices $D_1 \sim D_n$, $D_1 \sim D_n$ continue transmitting a beacon without the termination conditions of proximity information propagation. We define A termination conditions of a proximity information propagation as time passage t_1 since proximity information propagation started. When device $D_m (1 \leq m \leq n)$ receives beacon B, a PIP mechanism sends the received time t_{D-m} and the beacon information I_B to a dedicated server. The dedicated server checks the received time t connecting I_B . If t is not stored on the dedicated server, the dedicated server connects t_{D-m} and I_B and stores t_{D-m} as t. If t is stored on the dedicated server, we can estimate whether the time passed between t and t_{D-m} is t_1.

3.2 Device identification algorithm based on proximity information propagation mechanism.

This mechanism enables device grouping by gathering the devices in space. Device grouping recognizes which device belongs to a group via the group ID. Gathered devices are assigned the same group ID.

Fig. 3 shows an algorithm for the device recognition method. In the algorithm, the input is the beacon information I_B of a received beacon B and the output is the group ID. First, using get Current Group ID() in line 1, the mechanism gets the group ID of the group to which the device belongs. If the device does not belong to a group, the mechanism receives a blank value. Using get Identifying Beacon() in line 2, the mechanism obtains the beacon information when the device has a group ID and stores the beacon information to IB . With check Beacon (I_B,I_b in line 3, the mechanism compares I_B.uuid to I_b.uuid, I_B.uuid to I_b.uuid, and I_B.minor to I_b.minor. If the result of all comparisons is true, check Beacon (I_B,I_b) is also true. If check Beacon (I_B,I_b) is true or Ib is a blank value, the algorithm resumes the process after line 4. If I_B meets the proximity state condition, estimate Proximity (I_B.rssi) in line 4 is true and the algorithm continues the process in lines 5, 6, and 7. In line 5, advertise(I_B) generates the beacon B based on I_B and advertises the beacon B. In line 6, create Group ID (I_B) generates a group ID based on I_B. The generated group ID is stored as an output value. Then, create Group ID (I_B) converts I_B to the group ID. In line 7, store Identifying Beacon (I_B) stores

the I_B used to generate the group ID. The processes in lines 9 and 10 are executed if and only if the I_B used to generate the group ID meets the proximity state condition. The function stopAdvertise() in line 9 stops advertising the beacon, and remove Identifying Beacon() in line 10 removes the I_B that store Identifying Beacon(I_B) stored in line 7. In the case where the algorithm executes lines 9 and 10, the group ID is kept without being removed.

Input: I_B
Output: $groupID$
1: $groupID \leftarrow getCurrentGroupID()$
2: $I_b \leftarrow getIdentifyingBeacon(groupID)$
3: **if** $checkBeacon(I_B, I_b) \cup I_b$ is $empty$ **then**
4: **if** $estimateProximity(I_B.rssi)$ **then**
5: $advertise(I_B)$
6: $groupID \leftarrow createGroupID(I_B)$
7: $storeIdentifyingBeacon(I_B)$
8: **else**
9: $stopAdvertise()$
10: $removeIdentifyingBeacon()$
11: **end if**
12: **end if**

Figure 3. Device identification algorithm based on proximity information propagation machanism

3.3 Application of the Device Recognition Method with the Proximity Information Propagation Mechanism

In this section, we show an ad hoc login system using the device recognition method. This system executes device grouping using the PIP mechanism. The system assigns group ID to several devices using proximity information propagation. The devices assigned the group ID receive the authentication information to login using the group ID. The authentication information enables the devices to access the contents on a server.

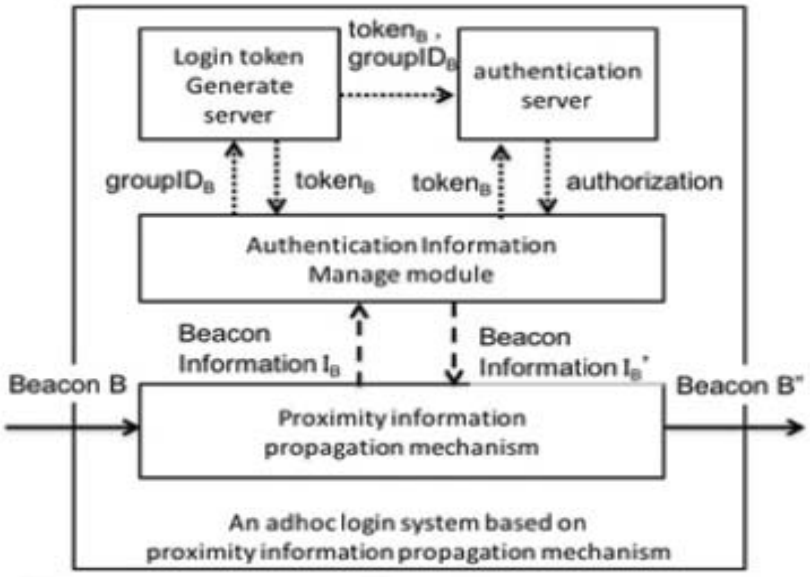


Figure 4. Authentication system based on proximity information propagation

Fig. 4 shows the system architecture. This system is composed of a PIP mechanism, an authentication information managing module that cooperates with the PIP mechanism, a login token generating server, and an authentication server. When the authentication information managing module receives the beacon information IB from the PIP mechanism, it generates the group ID using I_B.rssi, I_B.major, and I_B.minor. The authentication information managing module can then send the group ID to the login token generating server. When the login token generating server receives the group ID from the authentication information managing module, the login token generating server returns $token_B$ connecting the authentication information to the authentication information managing module. At the same time, the authentication information managing module sends the group ID and $token_B$ to the authentication server. The authentication server connects and stores the group ID, $token_B$, and the authentication information. The authentication information managing module sends the received $token_B$ to the authentication server, and if the authentication succeeds, the authentication server sends the authorization information to the authentication information managing module. This system can grant access to some content on the authentication server via the authorization information.

Figure 5. Login using the authentication system based on proximity information propagation

Fig. 5 shows a situation where the iOS applications on several devices login without being taken off a display shelf. The iOS applications on three devices on the left-hand side of Fig. 5 have completed login and have transited to another screen. We can prep systems composed of iOS applications for operation while the devices are in a storage space. Therefore, we can carry as many devices as we need and have additional devices getting ready and waiting to be used. In this way, the module can enhance the convenience of an iOS application.

4. Evaluation Experiment

When a number of devices are placed near other devices, the devices prevent beacon transmission. We placed a number of devices on top of each other as a condition in which a number of devices are near other devices. In this configuration, we measured the RSSI over a length of time t ($1 \leq t \leq 300$).

Fig. 6 shows the experimental environment. We placed seven labeled iPads (iPad Air) in the center of a table (0.8 m by 1.4 m). The iPads were labelled 1 to 7 from the top to the bottom. We placed a beacon transmitter on iPad No. 1 during the experiment. For the experiment, we used My Beacon [6] as the beacon transmitter. In this section, we refer to the beacon transmitter as BTR. The BTR did not transmit until the beginning of the experiment. At beginning of the experiment, we powered up the BTR manually. the experimental environment.

The result of the experiment can be seen in Fig. 7. The x-axis on the graph shows the time axis during the experiment, and the y-axis shows the RSSI of the beacon transmitted by BTR. The RSSI of iPad No. 1 was stable between -43 dBm and - 40 dBm. The proximity of BTR was Immediate at all times t. Because BTR was located immediately above No. 1, we can see that the beacon transmitted by BTR to the closely placed iPad No. 1, which acted as the beacon receiver, was undamped. The RSSI of iPad No. 2 was stable between -57 dBm and -56 dBm. iPad No. 2 was located directly below iPad No. 1, and the proximity of BTR was Near at all times t. Thus, iPad No. 1 attenuated the beacon transmitted by BTR at all times t. The RSSIs of iPad Nos. 3, 6, and 7 were less than those of other near iPads at all times t and unstable between -67 dBm and -62 dBm. The proximity of BTR was Near at all times t. The RSSIs of iPads No. 4 and 5 were less than those of the other iPads and unstable between -71 dBm and -66 dBm.

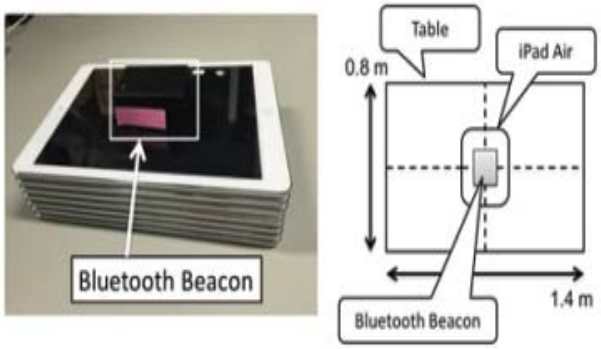


Figure 6. Experimental environment to measure the Bluetooth signal strength of piled beacon receivers

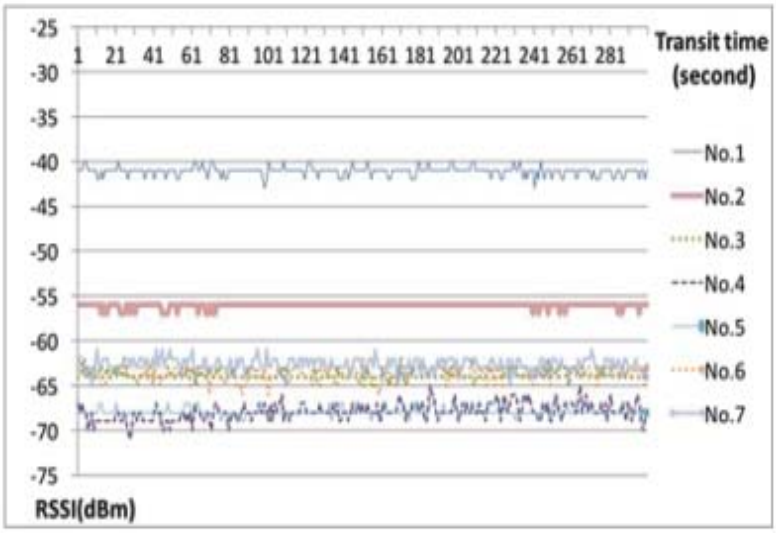


Figure 7. Experimental result without the proximity information propagation

The proximity of BTR that No. 4 received was Far at the times in the interval $0 \leq t_4 \leq 15$ and Near at other times. The proximity of BTR that iPad No. 5 received was Far at the times in the interval $0 \leq t_5 \leq 15$ and Near at other times. All the iPads were within a radius of 10 cm of the BTR; however, only iPad No. 1 had a proximity that was Immediate. From the above experimental result, we can see that if the devices are stacked, they will attenuate the receiving beacons from each other.

Next, we added the module to all iPads and measured the RSSI of beacons transmitted by the stacked devices during the time t ($1 \leq t \leq 300$). The experimental environment was the same as in Fig. 6. Fig. 8 shows the results of the second experiment. The RSSIs of beacons received by iPads No. 1 and 2 were stable between -35 dBm and -32 dBm. The RSSIs of beacons received by iPads No. 3 to 7 were unstable between -45 dBm and -36 dBm; however, they had larger values than the results in Fig. 7. The proximity of BTR for all iPads was Immediate. Therefore, we succeeded in making the seven stacked iPads detect at a proximity of Immediate using a beacon. In other words, we can achieve device grouping with a beacon by setting Immediate as the proximity call condition.

In addition, we placed stacks of iPad Air 2s as shown in the upper panel in Fig. 9 at the same table as Fig. 6. One stack of iPads was labeled No. 1 through 5, and other stack of iPads was labeled No. 6 through 10. In this condition, we conducted a preliminary experiment to examine the possibility of device grouping. iPads Nos. 1 and 6 were at the top of each iPad stack. The BTR was placed on iPad No. 1, 7 cm from the center of the screen. We measured the RSSIs of all the beacons that the 10 iPads received.

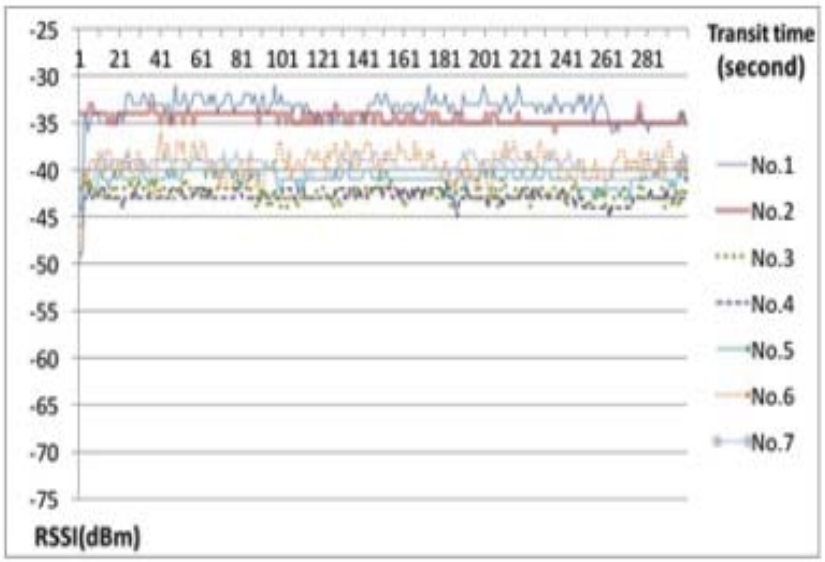
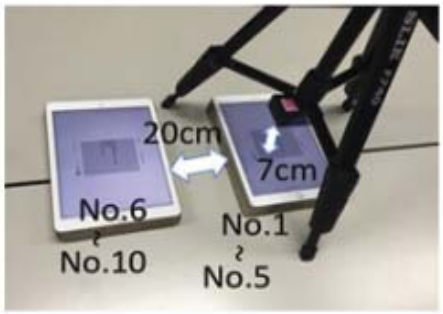


Figure 8. Experimental result with the proximity information propagation



	No.1	No.2	No.3	No.4	No.5	No.6	No.7	No.8	No.9	No.10	BTR
No.1		-36	-43	-43	-43	-52	-69	-62	-66	-63	-50.3
No.2			-44	-48	-49	-64	-84	-85	-80	-88	-57.3
No.3				-42	-46	-62	-83	-80	-81	-85	-54.9
No.4					-47	-66	-85	-87	-85	-88	-65.5
No.5						-62	-82	-80	-81	-85	-60.5
No.6							-36	-39	-42	-48	-52
No.7								-40	-45	-48	-62.1
No.8									-42	-42	-62.9
No.9										-40	-63.1
No.10											-63.3

Figure 9. Preliminary experimental environment and results with immediate information propagation to discriminate nearby groups

The lower table in Fig. 9 shows the result of the experiment. The table rows show the labels of each iPad as a beacon receiver. The table columns show the labels of each iPad as a beacon transmitter. The shaded table elements indicate where the proximity was *Immediate*.

If we simply implemented device grouping based on a proximity of Immediate, we would not be able to distinguish which group iPad No. 6 should be in. However, iPads No. 1 to 5 have Immediate proximity to each other and so do iPads No. 6 to 10. Therefore,

we need to define a Proximity value covering multiple beacon receiver proximities.

5. Conclusions

We implemented the PIP mechanism, which can group devices as a cluster in specific spaces and this paper presented the device identification algorithm based on PIP. The algorithm solved problems that devices can be obstacles against detecting a right group of stacked devices. This mechanism uses radio-transmitting and radio-receiving functions based on Bluetooth technology and transmits a beacon when the PIP mechanism meets a proximity ranging condition. By using the method, users can intuitively make a group of stacked devices as a cluster by transmitting a beacon in a specific area.

We showed that the PIP mechanism enables all the devices in a cluster to receive similar RSSI in the experiment described in 4 although receivers can be obstacles attenuating RSSI. Further, we showed that the PIP mechanism can distinguish each device among a number of devices aggregated in one place. We prepared a configuration in which a number of devices were stacked on top of one another as an example of a number of devices aggregated in one place. Then, we measured the RSSI. From the measurements, we concluded that we could group devices as a cluster in a situation where a number of devices were stacked.

In addition, we implemented a novel login system, which is convenient and secure. It is time consuming that all attendee at a conference securely share and input password. We build a login system based on the PIP mechanism that enables users to all log in to the iOS application as a group. People do not have to log in individually with each device. As a result, the amount of preparation time in circumstances such as a conference or a meeting is expected to be shortened. Moreover, it is secure because any password should not be shared.

Acknowledgement

This work was supported in part by JSPS KAKENHI Grant Number 15K00422, 16K00420.

References

[1] He, Zhiqiang., Cui, Binyue., Zhou, Wei., Yokoi, Shigeki. (2015). A proposal of interaction system between visitor and collection in museum hall by iBeacon, *In:* The 10th International Conference on Computer Science & Education (ICCSE), IEEE, 427-430.

[2] Kao, Hsin-Liu (Cindy)., Dementyev, Artem., Paradiso, Joseph A., Schmandt, Chris (2015). NailO: Fingernails as an Input Surface, *In*: The Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems. ACM, p. 3015-3018.

[3] Aumi, Md Tanvir Islam., Gupta, Sidhant Goel, Mayank., Larson, Eric., Patel, Shwetak (2013). Doplink: Using the doppler effect for multi-device interaction, *In:* The Proceedings of the 2013 ACM International Joint Conference on Pervasive and ubiquitous computing. ACM, 2013, p.583-586.

[4] Suzuki, R., Murase, T. Shiramatsu, S. Ozono, T. Shintani, T. (2013). On an Implementation of a Smart Signage System based on Tablet Devices. *Computer Software*, 30 (2) 176-190.

[5] Goel, Mayank., Lee, Brendan., Aumi, Md. Tanvir Islam., Patel, Shwetak., Borriello, Gaetano Hibino, Stacie Begole, James (2014). SurfaceLink: using inertial and acoustic sensing to enable multi-device interaction on a surface, *In:* The Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2014, p.1387-1396.

[6] MyBeacon. http://www.aplix.co.jp/product/mybeacon/mb004ac