

PAPER

Rhythm Tap Technique for Cross-Device Interaction Enabling Uniform Operation for Various Devices

Hirohito SHIBATA^{†a)}, Junko ICHINO^{††}, Shun'ichi TANO^{†††}, *Members,*
and Tomonori HASHIYAMA^{†††}, *Nonmember*

SUMMARY This paper proposes a novel interaction technique to transfer data across various types of digital devices in uniform a manner and to allow specifying what kind of data should be sent. In our framework, when users tap multiple devices rhythmically, data corresponding to the rhythm (transfer type) are transferred from a device tapped in the first tap (source device) to the other (target device). It is easy to operate, applicable to a wide range of devices, and extensible in a sense that we can adopt new transfer types by adding new rhythms. Through a subjective evaluation and a simulation, we had a prospect that our approach would be feasible. We also discuss suggestions and limitation to implement the technique.

key words: *rhythmical taps, ad-hoc network connections, cross-device interaction*

1. Introduction

We are currently using various kinds of digital devices from small ones (e.g., smartphones, laptop PCs, tablet PCs, e-book readers, and smart watches) to big ones (e.g., desktop PCs, projectors, printers, scanners, and wall-size displays). In this situation, it is desirable that we could easily transfer data among devices by establishing ad-hoc network connections. Then, it will be convenient when switching projected PCs in a meeting, sending files to all audiences during a discussion, printing documents with a printer outside a home office, searching Web pages using text sent by other devices, or exchanging personal information with people acquainted at a party.

To transfer data from one device to others, we must specify a *source device* and *target devices*. Moreover, it is desirable to allow specifying what kind of data should be transferred, where we call this *transfer type*. One might want to send a file, a page, a mail address, a URL, an image, or text, depending on the situation. Although troublesome procedures to setup network connections can be hidden from users by designing a good user interface, these three types of information (a source device, target devices, and transfer type) cannot be skipped to transfer data across digital devices.

Manuscript received February 26, 2019.

Manuscript revised June 25, 2019.

Manuscript publicized September 19, 2019.

[†]The author is with Fuji Xerox Co. Ltd., Kanagawa-ken, 259–0157 Japan.

^{††}The author is with Tokyo City University, Yokohama-shi, 224–8551 Japan.

^{†††}The authors are with The University of Electro-Communications, Chofu-shi, 182–8585 Japan.

a) E-mail: hirohito.shibata@fujixerox.co.jp

DOI: 10.1587/transinf.2019EDP7060

As a main target field, we aim to support intensive document work such as research, investigation, or intellectual property management. In these human intellectual activities, people organize their ideas by spreading out many documents, shifting their attention across them, and frequently referring and comparing large amounts of sources [1], [2].

Currently, workers often perform such work using only a single digital device such as a desktop PC or a mobile tablet PC. However, different devices have different strengths and different weaknesses. There is no best single device. The use of a PC with a keyboard will be convenient to edit a document, but to read it, it is useful to use an eye-friendly and light-weight slate with an electronic paper panel [3]. It seems to us that the current work environment that people perform tasks only with a single device looks as if people cook dinner using a Swiss army knife, where it sacrifices the usability to give many functions in a small device. If we could smoothly and easily coordinate various digital devices, people would work comfortably and efficiently by selecting appropriate devices depending on the task or situation or by using multiple devices simultaneously.

To achieve such a future, it is desirable that people can transfer data among various devices in a uniform manner. When we concentrate on document work, we often do not want to look away from a point where we are reading or thinking. Therefore, our cross-device interaction technique should not heavily rely on vision. Additionally, the support of intensive document work requires diversified coordination styles, and then the system should allow specifying various transfer types.

In this research, we propose a novel cross-device interaction technique that enables uniform operation for various devices with different size and different OSs and that allows users to specify various transfer types in addition to a source device and target devices. This paper reports our initial trials to examine the feasibility of our proposal*. Because our final goal is to support knowledge workers' intellectual document work, we mainly assume a situation where a single person can comfortably work using multiple devices which are within hands reach of the person as our first step. Additionally, a security issue is not our current main concern, although we discuss it later in this paper.

*This paper is a revised version of our previous one [4].

2. Related Work

Until now, various user interface techniques for cross-device data transfer have been proposed. They are roughly divided into four types.

The first ones are techniques to connect desktop spaces visually in OS level and enable to transfer data using a GUI operation like a drag-and-drop user interface [5]–[7]. This operation is intuitive, but we must connect desktop spaces by arranging two devices in parallel [5], [7] or sliding a pen across multiple displays [6] before transferring data. Additionally, most these techniques were developed to handle mobile devices. Since we must move devices to connect desktops, we cannot adopt these techniques for heavy devices or devices fixed to an environment. Moreover, it is difficult to implement a GUI interface across different OSs.

The second ones specify two devices by using special gestures such as moving devices closely each other [8], tossing devices [9], sliding a finger on a desk [10], moving a hand on a desk [11], and multi-user cooperative gestures [12]. They all provide an easy intuitive user interface. However, they need special sensors to detect the proximity or position of devices. Additionally, we cannot adopt most these techniques to unmovable devices which cannot be held by hands.

The third ones point to multiple devices by using special devices such as pens with IDs [13], fingers [14], or special buttons [15]. They can be applied to unmovable devices. However, they need additional devices such as pens, buttons, or sensors to detect a finger print. Their implementation cost is not low. We cannot expect that a wide range of devices provide such devices or sensors in the future.

The fourth techniques use synchronous gestures to specify the connection of devices [16]. In these techniques, synchronous events across devices such as shaking devices [17], bumping devices [18], and pressing and releasing buttons [19], are used as key gestures to connect the devices. They can be easily implemented without using any special sensors or special devices. However, synchronous shaking and bumping gesture cannot be applied to heavy devices or fixed devices. Moreover, users usually cannot specify various transfer types because available gesture types are restricted. In actual, they are applied for restricted transfer types such as exchanging email addresses.

As we described above, previous approaches do not satisfy our requirements. They do not cover a wide range of devices or do not have the scalability to specify various kinds of transfer types.

3. Framework: Rhythm Tap Technique

3.1 Basic Framework

We consider a solution based on synchronous gestures because they are easy to operate and they do not require any additional devices or special sensors.

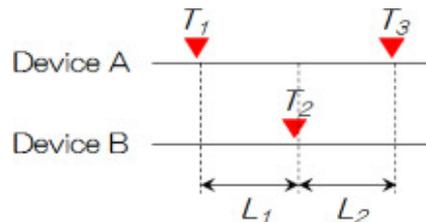


Fig. 1 A simple example of rhythmic taps.

In addition, we develop our technique using a touch gesture because touch sensors on panels are now widely used and we can assume that much more devices will provide touch sensors in the future. Moreover, touch interaction can be applicable in various devices from small ones to large ones and from mobile ones to fixed ones. Therefore, our approach will be applicable to various types of devices.

However, as we described before, previous techniques of synchronous gestures have a problem that they do not allow specifying various transfer types. They transfer data based on a fact that same event occurred at different devices simultaneously. To specify various transfer types, we must allow adding new gesture types.

To give the variety of events to synchronous tapping, we expand a single tap to a sequence of taps on a temporal axis. In our framework, we connect devices if rhythmic taps occur across multiple devices and transfer data from the one tapped at the first tap to the other, where we call this *rhythm tap technique*.

We can think up large number of rhythmic tap patterns. This means that we can specify various transfer types using different rhythms. Our framework is also extensible in a sense that we can add new transfer types by adding new rhythmic tap patterns.

Figure 1 shows a simple example of this framework. Let's consider the situation that tap events occurred with time T_1 , T_2 , and T_3 in devices A, B, and A in this order. Let $L_1 = T_2 - T_1$ and $L_2 = T_3 - T_2$. If $|L_1 - L_2|$ is small enough or $|L_2/L_1|$ is close to 1, then we consider this tap sequence is rhythmic and data corresponding to the rhythm are transferred from the device A to the device B.

Previous studies have observed or investigated what kind of gestures are performed for the interaction with multiple mobile devices or multiple displays [20]–[22]. However, we cannot find any description on rhythmic taps across devices. Since rhythmic taps do not seem to be an intuitive user interface, we must consider why we adopt the rhythm tap technique for data transfer.

First, almost all people can tap rhythmically for multiple devices, if the rhythm is not complex. Everybody can tap simple rhythms easily. This means they can easily specify transferring data among devices.

Second, it is difficult to perform rhythmic taps with other people. Let's think about the case of playing music. It is difficult to play a session with those who do not have an intention to cooperate with other people. This means it is difficult to steal data by breaking into other people's rhyth-

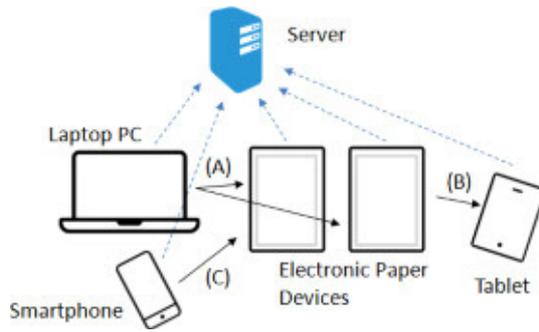


Fig. 2 A simple example of rhythmic taps.

mical session intentionally. We can expect users can control the speed of rhythmic taps to prevent from being stolen data.

Third, if there is a conductor and people have an intention to cooperate with others under the conductor, they can perform rhythmic taps together easily. This means users can transfer data to multiple users at the same time. For example, let's consider a situation that a presenter wants to send a file to all audiences in a meeting. In this case, the presenter may become a conductor, decide a rhythmic tap sequence (e.g., $T_1T_2T_3$ of Fig. 1), assign conductor's taps (e.g., T_1 and T_3 of Fig. 1) and audiences' taps (e.g., T_2 of Fig. 1), and perform rhythmic taps with audiences by keeping rhythm as a conductor. Then the presenter's file will be sent to audiences' devices. This means our framework can be used for one-to-many data transfer.

Finally, as we described before, rhythms are diversified. We can create innumerable new rhythmic tap sequences on a temporal axis. This means our framework can cope with various transfer types.

3.2 Use Scenario

In our framework, all digital devices used in the framework must be registered in the server in advance. Each device sends messages when tap events occur as shown in Fig. 2. If the server detects a registered rhythm tap pattern, the server sends a message to the source device, and the source device send the corresponding data to the target device.

We show an example of the use scenario in intensive document work. When you are working in a laptop PC, you may find a document that you want to read deeply. In such a case, when you rhythmically tap the laptop PC and a thin electronic paper device, the document is sent to the electronic paper device (Fig. 2 (A)). Then you can read the document comfortably with holding it with the light device. If you send another file to another electronic paper device, you can easily layout the documents to compare information of different sources.

If you want to search on a Web about a word of the document, you select the word and rhythmically tap the reading electronic paper device and a tablet device with another rhythm. Then the selected text is sent to the tablet

(Fig. 2 (B)) and Web search is performed using the text on it. You can easily refer to information without hiding the reading document.

Electronic paper devices are light and easy to handle while reading, but their page drawing is slow and it is difficult to jump to far distant pages [3]. In such a case, you can use a smartphone as a controller of electronic paper devices. When you find a desired page in the smartphone using an overview or quick scroll function, and when you rhythmically tap the smartphone and the electronic paper device, then the document and the page number are sent to the electronic paper device (Fig. 2 (C)). In this case, you can jump to other pages in the electronic paper device using the smartphone as a page navigation controller.

3.3 Overview of Studies

In our study, we focus on examining feasibility of our framework rather than implementation. We also focus on a user interface technique to specify what to transfer (i.e. transfer type) from a source device to target devices. Additionally, we have not decided how to assign rhythm patterns to ways of coordination. Our current concern is in how we can precisely detect intended rhythmic taps without detecting unintended spontaneous tap sequences.

We explored how to establish our framework in the following four steps. In the remaining of this paper, we introduce our studies in this order.

1. **Selecting rhythmic tap sequences.** We must understand what kind of rhythmic taps are preferred or easy to operate. We conducted subjective evaluation of rhythmic taps and selected some of them that seemed to be reasonable as instruction methods when transferring data.
2. **Collecting rhythmic tap patterns.** Users cannot perform rhythmic taps precisely. To understand the distribution of the accidental errors, we collected users' actual data of rhythmic taps.
3. **Creating detecting method.** We considered how to detect rhythmic taps based on the analysis of the users' actual rhythmic taps.
4. **Assessing the possibility of false detection.** If the detecting method detects many false detections, that is, if it detects users' spontaneous tap sequences by chance frequently, this method cannot be used in real world. Therefore, we conducted a simulation to check how many false detections occurred in our daily situation.

4. Step 1: Rhythmic Tap Sequences

To understand preference or easiness to operate of rhythmic taps, we conducted a subjective evaluation. The participants were 15 people (13 men, 2 women). Their ages were from 22 to 26 (avg. 23.8).

We selected 99 rhythmic taps, where the tap count of each rhythm was less than six. They are systematically selected with considering some features of rhythms such as tap

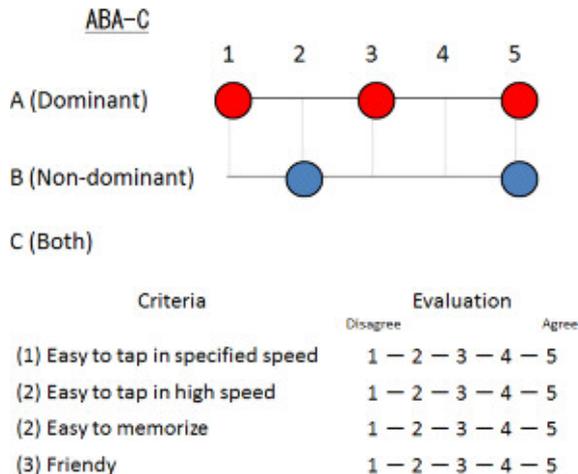


Fig. 3 A sample (ABA-C) of an evaluation sheet.

counts of dominant hand, the count of pause, and successive tap counts.

The participants evaluated each rhythmical taps in five-point scaling for four evaluation items: easiness to tap in specified speed (the interval of taps was 150 ms)[†], easiness to tap in a high speed, easiness to memorize, and friendliness.

Figure 3 is a sample of the evaluation sheets. As the notation of rhythms, “A” stands for a tap by a dominant hand, “B” stands for a tap by a non-dominant hand, “C” stands for a tap by both hands, and “-” stands for a pause. For example, the tap timing of “ABA-C” is depicted in Fig. 3.

For each rhythmical tap sequence, at first, the participants heard the rhythmical sequence as a sound created by a system with the specified speed on a Surface Pro 3. Next, they actually tapped it using their both hands on a desk without using any digital devices. People often cause errors when tapping rhythms quickly and we want to understand how people can easily tap rhythms in high speed. We required the participants to tap the rhythm in high speed. In this case, they were encouraged to tap it as fast as possible. Finally, they responded to the four evaluation items.

We evaluated rhythmical taps by average scores of the four evaluation items. Table 1 shows top 12 rhythms. We selected top eight rhythms (ABAB, ABA, ABABA, AB-AB, AB-C, AB-A, ABA-A, and AAB). Moreover, we added three rhythms (AB-B, ABA-C, and ABB) by considering the symmetry of both hands. ABB was ranked in 22 and its average score was 4.55.

From this analysis, we obtained following findings that work as suggestions to create user friendly rhythmical tap sequences.

- Consecutive taps of a single tap (i.e., A or B) and simultaneous taps (i.e., C) are difficult to tap. That is, tap sequences of AC, BC, CA, or CB are difficult to tap.
- A tap sequences is preferred if it includes a pause (i.e.

[†]It is nearly equal to 100 BPM on 16-beat of drums, which is a major rhythm in pop music.

Table 1 Results of subjective evaluation of rhythmical tap sequences. Top 12 rhythms and their average scores of the four evaluation items: (1) easy to tap in specified speed, (2) easy to tap in high speed, (3) easy to memorize, and (4) Friendly. Bold rhythms are selected ones.

Rank	Rhythm	Avg.	(1)	(2)	(3)	(4)
1	ABAB	5.00	5.00	5.00	5.00	5.00
2	ABA	4.95	4.93	5.00	5.00	4.87
3	ABABA	4.93	5.00	4.93	4.93	4.87
4	AB-AB	4.93	5.00	4.93	5.00	4.80
5	AB-C	4.83	4.93	4.73	4.87	4.80
6	AB-A	4.82	5.00	4.67	4.80	4.80
7	ABA-A	4.78	4.87	4.87	4.67	4.73
8	AAB	4.73	5.00	4.60	4.87	4.47
9	AABA	4.72	4.87	4.53	4.80	4.67
10	ABA-B	4.72	4.87	4.87	4.60	4.53
11	AB-B	4.70	4.87	4.67	4.60	4.67
12	ABA-C	4.70	4.93	4.60	4.67	4.60

“-”). However, if it includes consecutive two pauses (i.e. “--”), it is not preferred. We think that is because it is difficult to express two pauses which can be easily differentiated from a single pause.

- Simultaneous taps (i.e., C) are preferred if there is a pause before them. That is, the tap sequence “-C” is preferred.

5. Step 2: Collecting Rhythmical Tap Patterns

To collect users’ actual rhythmical taps, we asked participants to tap rhythms in various conditions.

Method The experimental design was a four-way factorial design.

The first factor was the *directions* of display surfaces with two levels: Horizontal and Vertical. We used Microsoft Surface Pro 3 as a tapping device. Participants were sitting on a chair. In the Horizontal condition, we detached a keyboard and put the device on a table horizontally. In this situation, they are tapping a horizontal panel slouchingly. In the Vertical condition, we put the display surface vertically by using the setup stand of the back of the device. In this situation, participants were tapping on vertical panel with their back ramrod-straight.

The second factor was *rhythms* with eleven levels: ABA, AAB, ABB, ABAB, AB-A, AB-B, AB-C, ABABA, ABA-A, ABA-C, and AB-AB. They all are selected in the previous section.

The third factor was *hands* of the first taps with two levels: Regular and Reverse. In the Regular condition, they start rhythmical taps with their dominant hand. In the Reverse condition, they start rhythmical taps with their non-dominant hand.

The fourth factor was *speeds* with four levels: Preferred, Slow, Fast, and Specified. In the Preferred speed, they tapped rhythmically with a speed they like. In the Slow speed, they tapped intentionally slowly. In the Fast speed, they tapped as fast as possible. In the Specified speed, at first they heard the taps in a specified speed (the interval of taps was 150ms) as a sound, and next they tapped at the same speed.

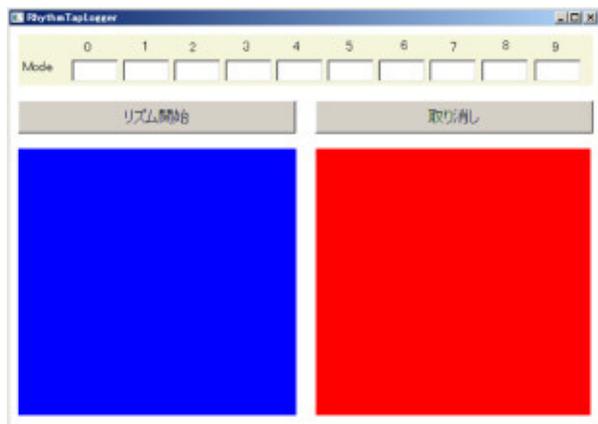


Fig. 4 A snapshot of an application to save logs of rhythmical taps.

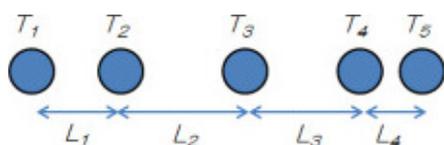


Fig. 5 Notation of taps and intervals.

The participants were 15 people (13 men, 2 women). Their ages were from 22 to 26 (avg. 23.6). They were all right-handed. All participants tapped rhythmical taps in all conditions. The order of conditions for each participant was randomized.

The participants tapped on the panel of Microsoft Surface Pro 3 (Windows 8.1). We used our own application to save tap logs. In this application, the right area was colored in red and the left area was colored in blue. The participants tapped the red area with their right hand (i.e., dominant hand) and tapped the blue area with their left hand (i.e., non-dominant hand), as shown in Fig. 4. In each condition, they tapped each rhythm at least five times. All tap events were saved in logs with timestamps.

An experimenter instructed the condition of tapping each time. The participants pushed a start button at first, and then tapped a given rhythmical tap sequence.

Notation Before discussing results of rhythmical taps, we explain notations. A tap is expressed as T_i as shown in Fig. 5. A tap T_i is also used as the time when the tap event occurred. To express a device name of T_i , we use a notation of device(T_i). L_i denotes the time interval between T_i and T_{i+1} (i.e. $T_{i+1} - T_i$).

Results We collected 9,717 rhythmical taps from the all participants. We removed rhythmical taps that the participants seemed to misunderstand the condition of tapping, e.g., tap count was different from that of a given tap sequence. We also removed outliers that one of the intervals of a tap sequence was outside the three standard deviations. As a result, we obtained 8,728 valid rhythmical taps.

Next, we analyzed the order of right taps and left taps. As a result, 8,678 rhythmical taps (99.4% for the valid ones) were correct ones, where the order of the right taps and the

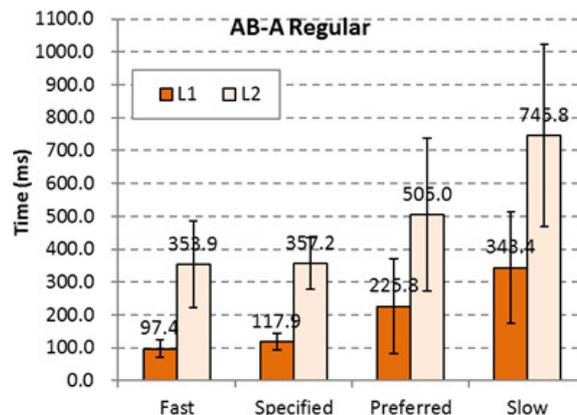


Fig. 6 Comparison of the first interval (L_1) and the second interval (L_2) of AB-A in the regular condition.

left taps was correct. There was no significant difference between conditions of all factors. For example, regarding the rhythms, the correct rate was the highest in the AB-C (100.0%) and the lowest in the ABAB (98.9%), but the difference was not significant. Regarding the speed factor, the correct rate was the highest in the Slow (99.5%) and the lowest in the Fast (99.3%), but the difference was not significant.

We can consider that the participants could tap rhythms in all conditions correctly enough. This indicates that our approach of rhythm tap technique works well as a user interface to transfer data.

We analyzed the correct 8,678 rhythmical taps (5.16 per condition and per participant) to determine the algorithm to detect rhythmical taps.

The tapping speed was fastest in the Fast, the Specified, the Preferred, and the Slow, in this order. According to one-way repeated ANOVA, the effect of the speed was significant ($F(3, 56) = 2.76, p < .001$). The average intervals of the first two taps were 115.4, 139.2, 217.1, and 391.6 ms in these conditions respectively.

Next, we focus attention on the length of a pause. We found that the participants took time for pauses relatively for a long time. Figure 6 compares the length of the first interval (L_1) and the second interval (L_2) of AB-A in the Regular tapping. Ideally, L_2 is a double of L_1 in all conditions, i.e., $L_2/L_1 = 2$. However, the values of L_2/L_1 were 3.63, 3.03, 2.23, and 2.17 in the Fast, the Specified, the Preferred, and the Slow, respectively. The participants tended to take longer time for a pause more than the theoretical length, and this tendency is more prominent when they tap fast.

We think that they took a longer break to emphasize the existence of a pause. It is a kind of deformation. If tap speed is fast, the length of tap intervals is hard to be recognized. Therefore, it might need a longer break to show the existence of a pause.

People's rhythmical taps are far from ideal ones and are not determined systematically based on the mathematical proportional relation. Considering this fact, to detect rhythmical taps, we cannot assume that people tap rhythms

precisely. We must determine how much degree we should accept the fluctuation of each tap according to rhythms.

6. Step 3: Detecting Method

Next, we consider how to detect rhythmical taps with considering the fluctuation of people's actual taps. In other words, we must find out an algorithm that discriminates whether a tap sequence is rhythmical or not.

Let's begin with considering simultaneous taps. The third tap and the fourth tap of AB-C must be tapped simultaneously ideally, and the fourth tap and the fifth tap of ABA-C must be also tapped simultaneously. Analyzing the difference of these participants' simultaneous taps, the upper 95% limit was 46.3ms (D_{max}). Therefore, we consider that two taps are tapped simultaneously if the time difference of these taps is less than this value.

Regarding the time interval of the first two taps of all rhythms, the lower 95% limit was 47.0ms (S_{min}) and the upper 95% limit was 451.3ms (S_{max}). If first two taps occur within this time interval (i.e. $S_{min} \leq L_1 \leq S_{max}$), new rhythmical taps might start from the two taps.

To detect rhythmical taps, we assume that all tap events have a device name and time. To consider a tap sequence as a rhythmical one, we use two criteria: *the validity of devices* and *the validity of intervals*.

Regarding the validity of devices, we check the order of tapped devices. For example, in rhythm ABA, the device of the first tap is different from the device of the second tap and is the same as the device of the third tap. That is, if a tap sequence of ABA is $T_1T_2T_3$, $device(T_1) \neq device(T_2)$ and $device(T_1) = device(T_3)$.

Regarding the validity of intervals, we check the length of intervals between taps. We estimate L_2 , L_3 , and L_4 based on L_1 . Table 2 shows estimation formulas for each rhythm obtained from the data of the previous section based on a single regression analysis.

If actual L_2 , L_3 , and L_4 are contained in the 95% prediction interval calculated by the estimation formulas and standard errors, we considered the tap sequence as a rhythmical one. The lower bound and the upper bound of the 95% prediction interval are calculated as $y_0 - 1.96 * SE$ and $y_0 + 1.96 * SE$, where y_0 is an estimation value and SE is the standard error.

The criteria to judge if a tap sequence is rhythmical or not are different for every rhythm. We present just one example. Regarding ABA-C, a tap sequence $T_1T_2T_3T_4T_5$ is considered as a rhythmical one, if these taps satisfy the following requirements.

- **The validity of devices:** $device(T_1) \neq device(T_2)$, $device(T_1) = device(T_3)$, $device(T_4) \neq device(T_5)$ and $(device(T_1) = device(T_4) \text{ or } device(T_1) = device(T_5))$.
- **The validity of the first interval:** $S_{min} \leq L_1 \leq S_{max}$.
- **The validity of the second interval:** Letting $[min_2, max_2]$ be 95% prediction interval of L_2 based on L_1 in ABA-C, T_3 is contained in $[T_2 + min_2, T_2 + max_2]$.

Table 2 Estimation of L_2 , L_3 , and L_4 based on L_1 for each rhythm. The values of the parentheses are standard errors.

Rhythm	Estimation of L_2	Estimation of L_3	Estimation of L_4
ABA	$0.91 L_1 + 37.2$ (52.7)		
AAB	$1.03 L_1 - 7.1$ (40.9)		
ABB	$0.85 L_1 + 43.7$ (37.4)		
ABAB	$0.96 L_1 + 14.1$ (27.0)	$0.95 L_1 + 10.6$ (25.7)	
AB-A	$1.54 L_1 + 183.0$ (89.1)		
AB-B	$1.48 L_1 + 195.9$ (80.1)		
AB-C	$1.45 L_1 + 207.9$ (92.9)		
ABABA	$0.93 L_1 + 22.5$ (28.0)	$0.97 L_1 - 0.5$ (21.6)	$0.92 L_1 + 36.4$ (34.5)
ABA-A	$0.97 L_1 + 28.0$ (27.2)	$1.54 L_1 + 166.5$ (76.9)	
ABA-C	$0.95 L_1 + 34.1$ (27.6)	$1.44 L_1 + 170.0$ (86.5)	
AB-AB	$1.59 L_1 + 146.5$ (63.0)	$0.95 L_1 + 0.3$ (22.9)	

- **The validity of the third interval:** Letting $[min_3, max_3]$ be 95% prediction interval of L_3 based on L_1 in ABA-C, T_4 is contained in $[T_3 + min_3, T_3 + max_3]$.
- **The validity of the final simultaneous taps:** $0 \leq L_4 \leq D_{max}$.

We implemented this algorithm and tried to detect rhythmical taps for the actual rhythmical taps collected in the previous experiment. As a result, this algorithm covered 84.15% of the participants' actual rhythmical taps.

7. Step 4: Possibility of False Detection

Next, we present a simulation to assess how many false detections occurred in a "natural" work situation.

We made a model of this "natural" work situation based on event logs of PC operations in our previous study [23]. This study was conducted for eight office workers of an intellectual property department, where most of their work was performed on PCs and they were heavy PC users. In the analysis, we considered that the participants' continuous PC work ended if any mouse or keyboard operation was not performed for more than two minutes. On average, they used PCs 11.70 times and for 3 hours 27 minutes a day in total. Each duration time of continuous PC use was 17.68 minutes. Clicks were performed 3.86 times per minute.

We create a model of our simulation based on the basic data of the previous study. In our simulation model, we assumed that all persons worked using five digital devices

Table 3 Results of simulation to measure how many false detections occur in a work situation.

Rhythms	Persons			
	1	5	10	20
ABA	3	11	57	233
AAB	2	15	44	193
ABB	1	44	205	1653
ABAB	0	0	1	7
AB-A	4	39	112	446
AB-B	6	107	497	3390
AB-C	0	0	0	0
ABABA	0	0	0	0
ABA-A	0	0	0	2
ABA-C	0	0	0	0
AB-AB	0	0	1	29

simultaneously instead of using a single PC as shown in our previous study. We also assumed that tapping in each digital device of our new model would be less than the clicking in the use of PCs in the previous study.

In the simulation model, we assumed that all persons owned five digital devices. They used these five devices simultaneously for 3 hours 27 minutes a day from 9:00 AM to 5:00 PM. The period of their work was five week days. Average duration time of the device use was 17.68 minutes and the taps on each device occurred 3.86 times per minute. The timing of digital device use and taps in each device were randomly selected.

In this model, each person used five digital devices simultaneously and each device was used at the same level as a single PC of the previous study. It seems that this setting of digital device use is very heavy in comparison with usual PC use.

Because tap events in each device occurred randomly, there must not be any rhythmical taps performed by a person intentionally. In the simulation, we try to detect rhythmical taps as false detections using the algorithm of the previous section. If many false detections are detected, this means that the algorithm is not secure.

We varied the number of persons (1, 5, 10, and 20) in the simulation. Table 3 shows the count of false detections in each condition for each rhythm. For ABAB, AB-C, ABABA, ABA-A, ABA-C, and AB-AB, which include at least four taps, no false detection was found for less than 10 persons. In the case of 10 persons (i.e., 50 devices), only one false detection was found in ABAB and AB-AB.

We can say that our detection method works enough to transfer data among devices in a small group with 10 persons using 50 devices, if rhythmical taps include at least four taps. In other words, we can find a field where our framework can be applied, if we select rhythmical taps with more than three taps.

8. Discussion

8.1 Application Fields

In the simulation to assess the possibility of false detection, we confirmed that our initial algorithm to realize the rhythm tap technique practically works well in a small group. However, this does not mean that it is secure. Rather, since there is a possibility that it causes unintentional data transfer in daily use, it is not desirable to adopt our framework to transfer highly-confidential data which are not definitely allowed to be transferred incorrectly.

However, there are some situations that such high security is not required. Personal intellectual activity is one of such examples. Rather, the strength of our framework is light-weight and convenient, that is, it is easy to operate, easy to implement with low cost, applicable to various devices, and extensible in adding new transfer types. The proposed framework will be effective in data transfer where rigid security is not required.

Our framework in current level can be used in a small group (less than ten persons). To limit the use of the framework to a small group, devices must be registered to a server in advance. Or we can register devices temporarily in a tentative group such as attendee of a meeting. As a way to register devices or as a way to create a tentative group, we can also adopt our rhythm tap technique again.

For example, we can temporarily register multiple devices in a same group by tapping the same rhythm together at the beginning of meetings or other gatherings. In this case, relatively long rhythms would be better to prevent from false detection for creating a group. Moreover, well-known and easy-to-tap rhythms would be better so that everybody can understand the rhythm and can avoid miss-tapping. In this case, tapping the same rhythm may also work as a way to bring a sense of togetherness and work as ice-breaking for a meeting.

To improve the security of our framework without restricting within a small group, we can provide two more alternative solutions. First, if we use long rhythms to transfer data, the false detection rate will decrease and our framework will become safer. Second, if we tap rhythms quickly, it is difficult for other people to break into the rhythmical tap session. This means that, by controlling the speed of rhythmical taps, we can reduce the risk that other people steal the data intentionally.

8.2 Future Work and Remaining Challenges

To check the validity of time intervals of a tap sequence, we predicted second and further tap intervals based on the first interval by using a single regression analysis. It is a very simple prediction method. We can adopt more accurate prediction method. Or we can adopt a discrimination analysis to detect rhythms by looking at whole taps simultaneously, not looking at each tap one-by-one like the method of this

paper. Trying to adopt such advanced methods, we expect that false detections would become fewer and our framework would become safer. In this paper, it is meaningful that we could have a prospect that our framework could work well by using even a simple prediction method.

To implement this framework, we need resolving some challenges described below. The first one is a problem of network delays. In this paper, we considered that there were no network delays to collect tap events from all devices. However, bluetooth or wireless LAN causes 1–20 ms delays to send data among devices.

As a next challenge, we should consider a peer-to-peer network architecture for an easy ad-hoc network connection. Our framework can be extended to a peer-to-peer network protocol. We need verifying it does not cause the increase of network traffic.

The proposed framework can be extended to detect collaborative rhythmical taps of one-to-many data transfer. However, it seems difficult to tap rhythmically with other people together. Therefore, we might need to lower the threshold to detect rhythmical taps if the system anticipates collaborative tapping. Additionally, it might be desirable that the system should give feedback to users whether their taps were too fast or too slow when they failed in collaborative rhythmical taps.

We must also think about user type. All participants in our experiments were in their 20's and they were all right-handed. We need gathering more diversified people including children, elder people, and left-handed people. Moreover, the preference of rhythms is strongly dependent on the culture. A rhythm preferred in one culture might be error-prone in other cultures. We must consider such a cultural aspect to implement the framework in a practical level. Additionally, investigation of cultural difference for preferred rhythms or easy-to-tap rhythms is also an interesting research theme.

Finally, we discuss the limitation of our framework. The rhythm tap technique can be applied to all devices that allow tapping, but we cannot apply this framework to a device refusing tapping such as a fragile device or a cup with water in it.

Additionally, our current framework is customized to a situation where a single person taps multiple devices which are within hands reach. To transfer data among distant devices or remote devices, we need other frameworks. Collaborative tapping of multiple users is a simple solution. For a single person to transfer data among distant devices, we can consider a framework that allows a time lag of rhythmical taps. For example, to send data among distant devices, we can allow that a user rhythmically taps in a source device and taps the same rhythm in a target device later within a certain short period of time.

9. Conclusion

This paper proposed a novel user interface technique to specify transferring data across digital devices. In this

framework, users specify what to transfer from a source device to target devices by tapping multiple devices rhythmically.

In the first experiment, we conducted a subjective evaluation of rhythmical taps and selected 11 rhythms. In the second experiment, we collected participants' actual rhythmical taps for the 11 rhythms. By analyzing the logs of rhythmical taps, we set up a reasonable method to detect rhythmical taps. In this method, false detection did not occur frequently (at most once in a week) in a small group of people (less than ten people) for rhythms including four or more taps. Although it remains many challenges to make this framework practical, we had a prospect that it could be effective in a small group use in a situation where high security is not required. Additionally, we obtained some practical suggestions to implement the framework of the rhythm tap technique.

As the future work, we need improving our algorithm by using more sophisticated prediction methods or discrimination analysis. Additionally, we need implementing this framework in real devices and comparing the effectiveness with other cross-device interaction techniques.

Trademarks

- Microsoft, Windows, and Surface are trademarks or registered trademarks of Microsoft Corporation.
- All brand names and product names are trademarks or registered trademarks of their respective companies.

References

- [1] A.J. Sellen and R.H. Harper, *The myth of the paperless office*, The MIT Press, 2001.
- [2] K.P. O'Hara, A. Taylor, W. Newman, and A.J. Sellen, "Understanding the materiality of writing from multiple sources," *International Journal of Human-Computer Studies*, vol.56, no.4, pp.269–305, Elsevier, 2002.
- [3] H. Shibata, Y. Fukase, K. Hashimoto, Y. Kinoshita, H. Kobayashi, S. Nebashi, M. Omodani, and T. Takahashi, "A proposal of future electronic paper in the office: Electronic paper as a special-purpose device cooperating with other devices," *ITE Transactions on Media Technology and Applications*, vol.4, no.4, pp.308–315, 2016.
- [4] H. Shibata, J. Ichino, T. Hashimoto, and S. Tano, "A rhythmical tap approach for sending data across devices," *Proc. MobileHCI '16*, Poster, pp.815–822, 2016.
- [5] P. Tandler, P. Prante, C. Müller-Tomfelde, N. Streitz, and R. Steinmetz, "ConnecTables: Dynamic coupling of displays for the flexible creation of shared workspaces," *Proc. UIST '01*, 2001.
- [6] K. Hinckley, G. Ramon, F. Guimbretiere, P. Baudisch, and M. Smith, "Stitching: Pen gestures that span multiple displays," *Proc. AVI '04*, pp.23–31, 2004.
- [7] N. Marquardt, K. Hinckley, and S. Greenberg, "Cross-device interaction via micro-mobility and f-formations," *Proc. UIST '12*, pp.13–22, 2012.
- [8] J. Rekimoto, Y. Ayatsuka, and M. Kohno, "SyncTap: An interaction technique for mobile networking," *Proc. MobileHCI '03*, 2003.
- [9] K. Yatani, K. Tamura, K. Hiroki, M. Sugimoto, and H. Hashizume, "Toss-It: Intuitive information transfer techniques for mobile devices using toss and swing actions," *IEICE Transactions on Information & Systems*, vol.E89-D, no.1, pp.150–157, 2006.

- [10] K.-Y. Chen, D. Ashbrook, M. Goel, S.-H. Lee, and S. Patel, "Air-Link: sharing files between multiple devices using in-air gestures," *Proc. UbiComp '14*, pp.565–569, 2014.
- [11] R. Radle, H.C. Jetter, N. Marquardt, H. Reiterer, and Y. Rogers, "HuddleLamp: Spatially-aware mobile displays for ad-hoc around-the-table collaboration," *Proc. ITS '14*, 2014.
- [12] M.R. Morris, A. Huang, A. Paepcke, and T. Winograd, "Cooperative gestures: Multi-user gestural interactions for co-located groupware," *Proc. CHI '06*, pp.1201–1210, 2006.
- [13] J. Rekimoto, "Pick-and-drop: A direct manipulation technique for multiple computer environments," *Proc. UIST '97*, pp.31–39, 1997.
- [14] A. Sugiura and Y. Koseki, "A user interface using fingerprint recognition: Holding commands and data objects on fingers," *Proc. USIT '98*, pp.71–79, 1998.
- [15] Y. Iwasaki, N. Kawaguchi, and Y. Inagaki, "Touch-and-Connect: A connection request framework for ad-hoc networks and the pervasive computing environment," *Proc. PerCom '03*, pp.20–29, 2003.
- [16] G. Ramos, K. Hinckley, A. Wilson, and R. Sarin, "Synchronous gestures in multi-display environments," *Human-Computer Interaction*, vol.24, no.1-2, pp.117–169, 2009.
- [17] L.E. Holmquist, F. Mattern, B. Schiele, P. Alahuhta, M. Beigl, and H.-W. Gellersen, "Smart-Its friends: A technique for users to easily establish connections between smart artefacts," *Proc. UbiComp '01*, vol.2201, pp.116–122, 2001.
- [18] K. Hinckley, "Synchronous gestures for multiple persons and computers," *Proc. UIST '03*, pp.149–158, 2003.
- [19] J. Rekimoto, Y. Ayatsuka, H. Kohno, and H. Oba, "Proximal Interactions: A direct manipulation technique for wireless networking," *Proc. INTERACT '03*, pp.511–518, 2003.
- [20] E. Kurdyukova, M. Redlin, and E. André, "Studying user-defined iPad gestures for interaction in multi-display environment," *Proc. IUI '12*, pp.93–96, 2012.
- [21] T. Seyed, C. Burns, M.C. Sousa, F. Maurer, and A. Tang, "Eliciting usable gestures for multi-display environments," *Proc. ITS '12*, pp.41–50, 2012.
- [22] K. Takashima, T. Oyama, Y. Asari, E. Sharlin, S. Greenberg, and Y. Kitamura, "Study and design of a shape-shifting wall display," *Proc. DIS '16*, pp.796–806, 2016.
- [23] H. Shibata, "Measuring the efficiency of the introduction of large displays and multiple displays," *IPSJ Journal*, vol.50, no.3, pp.1204–1213, 2009. [in Japanese]



Junko Ichino received M.E. from University of Electro-Communications in 1998, and Dr. of Eng. from Kobe University in 2007. She is currently a professor of Tokyo City University. Her research interests include Human-Computer Interaction, Computer Supported Cooperative Work. She is a member of ACM, IPSJ, and IEICE.



Shun'ichi Tano is a Professor at Graduate School of Information Systems, University of Electro-Communications in Japan. He received the Doctor Degree in System Science from Tokyo Institute of Technology in Japan. He was the researcher in the Hitachi Ltd., the Laboratory for International Fuzzy Engineering Research, CMU and MIT. His research themes are twofold, intelligent system and human-computer interaction.



Tomonori Hashiyama is a Professor at Graduate School of Information Systems, University of Electro-Communications in Japan. He received Dr. of Eng. from Nagoya University in 1996.



Hirohito Shibata received M.Sci. from Osaka University in 1994, and Ph.D. in Eng. from the University of Tokyo in 2003. He is currently a Senior Research Principal at Research and Technology Department, Fuji Xerox Co. Ltd. He is also a part-time lecturer at Tokyo University of Technology and Otsuma Women's University. His research interests include human-computer interaction and cognitive science.