Privacy-Aware Communication for Smartphones Using Vibration

Inhwan Hwang, Jungchan Cho, and Songhwai Oh
CPSLAB, ASRI
School of Electrical Engineering and Computer Science
Seoul National University, Seoul, Korea
Email: {inhwan.hwang, jungchan.cho, songhwai.oh}@cpslab.snu.ac.kr

***Abstract*— We propose a novel communication method between smart devices using a built-in vibrator and accelerometer. The proposed approach is ideal for low-rate private communication and its communication medium is an object on which smart devices are placed, such as tables and desks. When more than two smart devices are placed on an object and one device wants to transmit a message to the other devices, the transmitting device generates a sequence of vibrations. The vibrations are propagated through the object on which devices are placed. The receiving devices analyze their accelerometer readings to decode incoming messages. Unlike radio-based wireless communication where eavesdropping of private communication is possible without the knowledge of the user, the proposed method can guarantee privacy as long as the object used for communication is secured. The proposed method is implemented on Android smartphones and comprehensive experiments are conducted to show its feasibility.**

I. Introduction

With the advent of a smartphone and the growth of smartphone users, short-range wireless communication between smartphones and between a smartphone and other devices has become an important capability of smartphones. A number of short-range wireless communication methods have been developed and embedded into smartphones, including Wi-Fi, Bluetooth, and Near Field Communication (NFC). However, the currently available short-range radio communication methods cannot guarantee complete privacy since eavesdropping is possible without the knowledge of the user. In this paper, we propose another short-range wireless communication method which utilizes the built-in vibrator and accelerometer in smartphones. The proposed method is motivated by the old-fashioned dot-and-dash communication method or Morse code and it can guarantee privacy in the presence of the user.

We assume that the communicating smart devices are placed on a single object, such as a table or a desk. The transmitting device generates a sequence of vibrations which is similar to the Morse code. The generated vibrations are propagated to the other devices through the object on which the devices are placed. The receiving devices decode the incoming message based on their accelerometer readings. The proposed method can be used to communicate with devices with accelerometers and no radio is required. Hence, the proposed method enables communication with a wide range of everyday objects which lack radio functionality, closing the gap between the cyber world and the physical world.

Compared to the existing short-range wireless communication methods, such as near field communication (NFC), radio-frequency identification (RFID), Wi-Fi, Zigbee (IEEE 802.15.4) and Bluetooth, the proposed method has several desirable characteristics. NFC is used for short-range communication with a maximum communication range of $20\,cm$ based on the RFID technology [3]. While a number of smartphone manufacturers are starting to package NFC chips in their smartphones, NFC is not yet widely available. But almost all smartphones that are available today are equipped with an accelerometer and a vibrator. In addition, NFC enjoys privacy at the cost of a very short communication range (at most $20\,cm$). The proposed method can provide a longer communication range than NFC as demonstrated in the present paper. RFID is another wireless technology for non-contact communication using radio [7] and suffers from the same shortcomings of NFC. While Wi-Fi [1], Bluetooth [4], and Zigbee [2] are widely available for longer range communication than NFC, eavesdropping without the knowledge of the user is possible [8] and private communication cannot be guaranteed. Komine et al. [5] proposed a new communication method using LED lights. This has an advantage in the sense of privacy over the other radio-based wireless communication method. However, it requires a line of sight between communicating devices and one-to-many communication is not possible without an extra apparatus.

As an alternative to the discussed radio-based wireless communication methods, Yonezawa et al. [9] proposed a method to transfer data from a smartphone to a notebook by generating vibrations from a smartphone. It is assumed that a smartphone is directly contacted to a notebook for communication [9]. In [10], the authors applied the method from [9] for pairing devices. Again, two communicating devices are required to be directly contacted with each other. Our work is similar to [9] but our method does not require a direct contact between devices and two-way communication is possible in ours. Since the vibration energy is dissipated in the object on which communicating devices are placed and there are vibration echoes as vibrations bounce back from the boundaries and other objects, a more sophisticated decoding method is required in our case.

The proposed method is ideal for low-rate private communication. As its communication medium is an object on which devices are placed, communication over a longer distance and broadcasting to multiple devices are possible. Furthermore, eavesdropping is not possible unless a foreign device is placed on the same communication medium object. But such foreign device can be easily identified by the user. These features distinguish the proposed method from [9], [10] and ratio-based communication methods. For example, a private radio-based communication session can be initiated by exchanging encryption keys using the proposed method. In addition to private communication among devices, another potential application of the proposed method is the task of configuring a large number of devices simultaneously without using radios. For instance, consider a desk which is used by multiple users. Suppose that a number of electronics, such as a notebook, a digital alarm clock, a tablet, an MP3 player, a telephone, and a TV, are placed on the desk and each electronic device on the desk has predefined user settings. If each user has her preferred settings for all electronics stored on her smartphone, then the proposed method can be used to configure all electronics on the desk for the user by simply placing the user's smartphone on the desk.

The remaining of this paper is organized as follows. The basic concept of the proposed method is described in Section II. Section III discusses issues when designing the proposed method. The implementation of the proposed method is given in Section IV and results from experiments are discussed in Section V.

II. Vibration-Based Communication

A. Overview

Figure 1 shows an overview of our system. We consider the problem of sending and receiving text messages between devices in this paper. But the proposed method can be applied to send packet-based messages of any type. When a user enters a string on her smartphone and press the *Send* button, the entered text is converted into a sequence of ASCII codes and then converted into a sequence of corresponding binary codes (each ASCII code is decoded using seven bits). The user's smartphone then vibrates based on each binary code in the sequence, transmitting each bit in the code. The receiving devices sample accelerometer readings to decode incoming messages.

B. Accelerometer Calibration

Calibration should be performed first due to the sensor variation of receiving devices. We need to convert raw accelerometer data into standard G unit using device-specific parameters. We get this calibrated accelerometer data by following the normalization scheme proposed in [6].

Let $\vec{a} = (a_x, a_y, a_z)$ be a raw accelerometer reading and $\vec{n} = (n_x, n_y, n_z)$ be the normalized accelerometer reading along each axis in unit G. The following function is used in normalization:

$$f(n_x, n_y, n_z) = \sqrt{n_x^2 + n_y^2 + n_z^2} \quad (1)$$

$$n_{axis} = K_{axis} a_{axis} + b_{axis}, \quad (2)$$

where K_x, K_y, K_z are the respective scaling factors and b_x, b_y, b_z are the offsets of the accelerometer. When the phone is stationary, the function f is assumed to be one. Hence, to find the normalized accelerometer readings n_x, n_y, and n_z, we need to estimate parameters K_x, K_y, K_z, b_x, b_y, and b_z

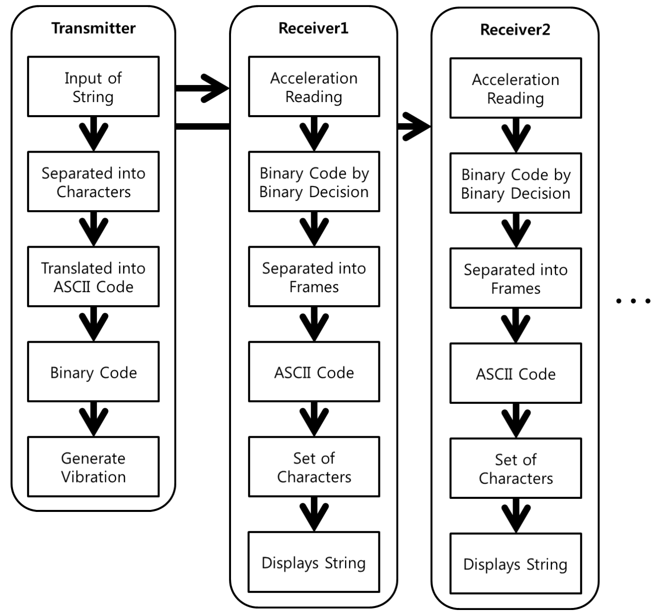


Fig. 1. An overview of the proposed system.

which make the function f unity when the phone is stationary. To solve this parameter estimation problem, we use a least square estimator based on the linear approximation of function f [6]. To get these parameters, we collected 2,000 accelerometer data in $\pm x$, $\pm y$, and $\pm z$ direction while the smartphone is not moving. The normalization parameters are computed as $\hat{K}_x = 0.0983, \hat{K}_y = 0.1024, \hat{K}_z = 0.1001, \hat{b}_x = 0.0222, \hat{b}_y = -0.0057$, and $\hat{b}_z = -0.0093$.

C. Transmitter Design

When a user enters a string, the transmitter separates the string into a set of characters and converts each character into a sequence of ASCII codes. Each ASCII code is then converted to a sequence of binary codes of seven bits. One bit of information is transmitted within a *frame* and a *superframe*, a collection of frames, is used to transmit an ASCII code. Figure 2 shows the structure of a superframe. Each superframe starts with a beacon bit which is used to synchronize devices. A superframe consists of 11 frames: beacon (1 frame), active frames (7 frames), and inactive frames (3 frames). The active frame field contains an ASCII code converted into a sequence of binary bits that it is made up of seven frames. The inactive frame field is a set of frames between the end of the active frame field and the start of the next superframe. The inactive frame field is used to reduce the time synchronization error which will be discussed in Section III-A.

In our implementation, each frame is one second long and this is due to the current limitation of the Android platform. Currently, stable vibrations are not possible on the Android platforms we have tested, if the time between vibrations is longer than one second. While the overall communication time is slow in our prototype, we believe the problem can be addressed if more sophisticated vibrators available in future smartphones. In order to transmit a bit `1`, we generate

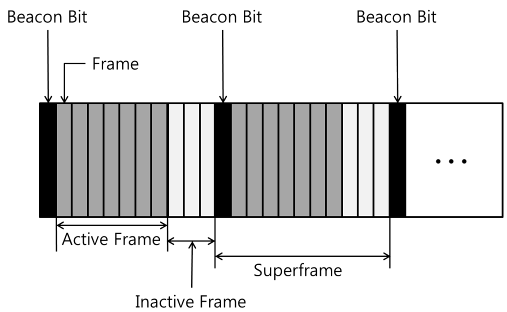


Fig. 2. The structure of a superframe.

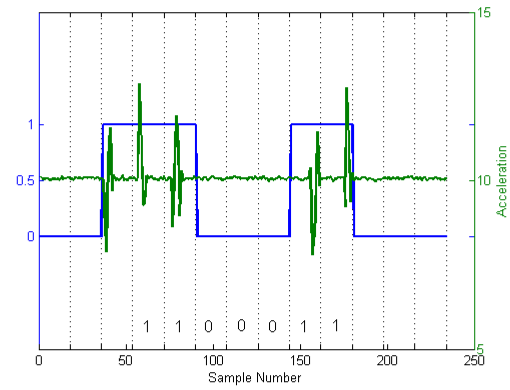


Fig. 3. The x-axis indicates the successive input acceleration readings. The y-axis on the left side indicates the calculated binary signal for each frame and the right side shows the raw data value of acceleration.

vibration for 0.2 second at around the center of a frame. No vibration within a frame means a transmission of 0. The inactive frames do not contain vibrations. We transmit a short vibration impulse for 0.2 seconds in order to minimize the effect of echoes as discussed in Section III-C.

D. Receiver Design

The value of f represents the magnitude of the normalized accelerometer reading. The magnitude of a normalized accelerometer reading is compared to the sample standard deviation to determine the presence of vibration. The recovered binary value for the i-th frame is determined as follows:

$$B(i) = \begin{cases} 1 & \text{if } \alpha \frac{1}{N} \sum_{j=1}^{N} \|\vec{n_{ij}}\| > \sigma_i \\ 0 & \text{if } \alpha \frac{1}{N} \sum_{j=1}^{N} \|\vec{n_{ij}}\| < \sigma_i, \end{cases} \tag{3}$$

where N is the number of samples in the i-th frame, $\|\vec{n_{ij}}\|$ is the normalized acceleration reading of the j-th sample in the i-th frame, and σ_i is the standard deviation of $\vec{S_i} = \{\|\vec{n_{i1}}\|, \|\vec{n_{i2}}\|, \cdots, \|\vec{n_{iN}}\|\}$.

Figure 3 shows an example with raw acceleration readings and detected binary bits for a single superframe. In Figure 3, each frame consists of 18 samples because the sampling rate considering energy efficiency of the phone used in the experiment (Samsung Galaxy S2) is approximately 18Hz which is one of sampling frequencies offered by Android. For the actual implementation, the number of samples within a frame is about 16 due to the computation time required to process binary streams.

With this basic concept of movement detection, we designed a communication tool using vibration. When this movement detection is well performed, we can intuitively understand how to transmit the signal with a vibrator and decode the transmitted signal using an accelerometer. However, we have encountered the following issues discussed in the next section when implementing the proposed method.

III. Design Considerations

We have encountered and addressed a number of issues when developing the proposed system and they are time synchronization, inconsistent sampling frequencies, minimizing the effect of echoes, detection threshold, and calibration issues.

A. Time Synchronization

Time synchronization between a sending device and a receiving device is essential for reliable communication. The wall clock available in smart devices are not sufficient since there is a delay from vibration propagation. In order to notify the receiving device that there is an incoming message, a beacon signal is used to synchronize devices. A beacon signal is encoded in a single bit. Hence, to send a single ASCII code of seven bits, eight bits are used. An example is shown in Figure 3. After a bit is detected by a receiver, it starts decoding each frame and find a bitstream of `1100011` whose ASCII code is the alphabet `c`. The use of beacon signal also prevents error accumulation as explained in Section III-B.

In addition to a beacon bit, an inactive period is added to the superframe as shown in Figure 2 to cope with the situation when a beacon bit is not correctly detected. When a beacon bit is missed by a receiving device, a normal bit will be treated as a beacon bit. We can better detect a beacon bit if the inactive period is longer. But a longer inactive period will increase the transmission time. After a number of experiments, we have found that an inactive period of three frames is a good trade-off between speed and correctness.

B. Inconsistent Sampling Frequency

Bit errors may appear if the transmission rate at the transmitter and the sampling rate of the receiver is misaligned. An example is shown in Figure 4. We transmitted successive `c` from the sending device and examined the received data from the receiving device. Note that the beacon bit and inactive frames are not used in this experiment.

The first seven bits are a correct result corresponding to the given input signal `c` (`1100011`). On the other hand, the next seven bits are `1110001` which is the alphabet `q`. The ASCII codes for `c` and `q` are `1100011` and `1110001`, respectively. We easily verify that the binary ASCII code of `q` is equal to one bit right-shifted binary code of `c`. This implies that small negligible misalignment can generate a bit size error when it is accumulated. As shown in Figure 4, the forth vibration spread from the last frame of first character

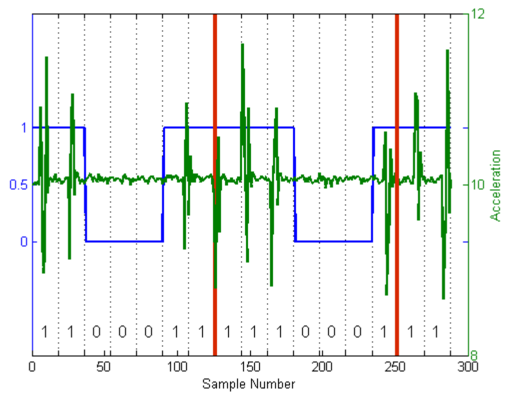


Fig. 4. The result of the signal received in the absence of beacon bit and inactive frames. Time synchronization is failed and there is a decoding error.

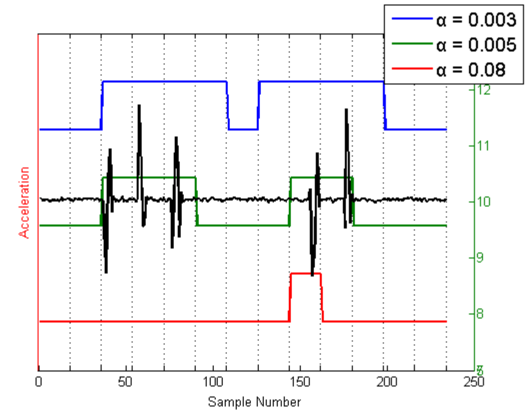


Fig. 5. The result varying with a different threshold α. Sensitivity is high when the threshold is low.

to the first frame of second character. So, we concluded that an initialization at every two characters can prevent the error accumulation. We solved this problem by introducing the beacon bit and inactive frames between superframes to improve synchronization among devices. We also reduced the number of vibration samples within a frame to reduce the effect of the inconsistent sampling rate.

Android-based smartphones supports a sampling rate between 17Hz and 18Hz for accelerometer readings. But when the sampling rate is set to 18Hz, the actual sampling rate becomes lower due to the CPU time required to process raw accelerometer readings. Hence, the size of each frame is adjusted to 16 samples in our implementation.

C. Minimizing the Effect of Echoes

As shown in Figure 3 and Figure 4, it is easy to observe that the peak of raw acceleration data at the receiving device is detected only for a short period less than one second. This can minimize the effect of echoes of the medium where the smartphone is placed. If the smartphone vibrates the medium, medium makes the receiving device to move. Compared to previous method, [9], [10], our proposed method is focused on distant communication via a rigid material that we need to consider the time delay between the sending device and the receiving device and the echoes from the medium. Empirically, we have find that transmitting a vibration for 0.2 second in a single frame can reduce the effect of echoes.

D. Overcoming Sensitivity Issues

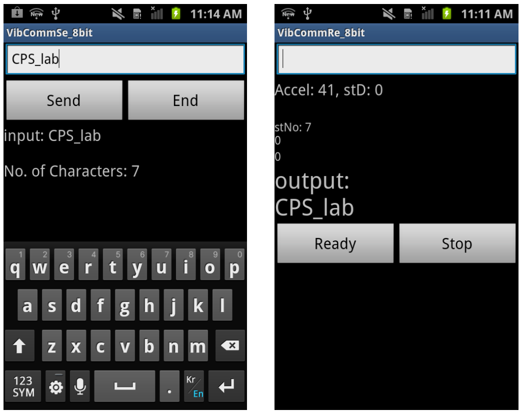
The next issue that requires an attention is the sensitivity of a receiving device and the vibration strength of a sending device. Sometimes the receiving device cannot detect a movement due to the different sensitivity of an accelerometer in the smartphone or the circumstantial factors such as the rigidness of the medium and interference resulted from outer forces. The problem resulted from the rigidness of the medium can be solved in some degree by adjusting the threshold α in (3). As shown in Figure 5, the threshold affects the bit detection described in Section II-D. The result shown in Figure 5 is obtained from a rigid wooden table and the distance between the sending device and the receiving device is $30\,cm$. As the threshold is getting larger, the movement detection is getting less sensitive. When it is less sensitive, it is robust against unstable surfaces and disturbances. However, less sensitiveness implies that there is a chance that the receiving device ignores the movement of the communication medium object. Hence, deciding the threshold value is important for applying the method to different types of communication medium objects and disturbances.

E. Calibration Issues

When we compared two smartphones of the same model, accelerometer readings were different for the same vibration sequences and the strength of vibration generated by each device was also different. Hence, in order to make our proposed method practical, it is necessary to adjust the vibrating strength of sending devices and calibrate the accelerometer for each devices. We believe this can be solved more easily as more sophisticated vibrators and accelerometers are incorporated into smartphones and they can be better calibrated from the factory. However, a fine calibration may be still required due to slight potential variations in those devices. In this paper, we have manually calibrated all smartphones used in experiments and focused on the communication mechanism.

IV. Implementation

Based on the method described in previous sections, we developed an Android application. Figure 6 shows a prototype of the developed Android application. The application has two sub-applications: sending and receiving applications. The sending application takes user's input as strings and convert them into characters (see Figure 6(a)). The characters are converted into seven-bit ASCII codes. Finally, the sending application constructs a superframe for each ASCII code and generates vibrations. On the other hand, the receiving application samples the acceleration readings when the user presses the *Ready* button (see Figure 6(b)). The receiving application starts detecting the movement of the medium using the method described in Section II-D and decodes the accelerometer readings into binary streams. The



(a) Sending Application (b) Receiving Application

Fig. 6. A prototype of the proposed system.

decoded binary bits are converted into ASCII codes based on the structure of the superframe.

The sampling rate provided by Android is given as a parameter and varies between model. Hence, we have empirically computed the sampling rate for each parameter value and they are approximately 6Hz, 18Hz, 50Hz, and 100Hz. Since the vibration transmission rate is low, a low sampling rate is sufficient in our case and it also reduces energy consumption. However, the sampling rate of 6Hz turns out to be too low as it misses vibrations. Hence, the next lowest sampling rate of 18Hz is used in our prototype.

V. Experimental Results

For experiments, the test string shown in Figure 7 is used. The threshold α is set to 0.005 in all experiments unless specified otherwise. We tested our method on following various situations by changing the communication medium objects, threshold, and distances between the transmitter and received.

A. Communication Medium Objects

We first tested our system using diverse communication medium objects, including a wooden table, a wooden table with a user using computer, a metal shelf, a plastic shelf, a cushioned chair, and a stack of paper. For all experiments, the test string shown in Figure 7 is transmitted repeatedly. In this experiment, the distance between a transmitter and a receiver is $50cm$.

abcdefghijklmnopqrstuvw
xyzABCDEFGHIJKLMNOPQ
RSTUVWXYZ0123456789~
!@#$%^&*()CPSLABASRIS
NUCPSNA2012HWANG

Fig. 7. A test string used in the experiment. There is a total 100 characters.

We tested on six different types of conditions by varying the material of the communication medium and interference. The accuracy of reconstructed strings at the receiver for each situation is shown in Table I. Each trial was done three times so that the total number of characters transmitted was 300. We counted the number of correct characters at the receiving device and the accuracy was more than 95% for all cases. As shown in Table I, even though the threshold α was set for a wooden table, the average accuracy using other communication medium objects was 96%. We can further improve the accuracy by adjusting the threshold for each situation and employing error-correction coding schemes such as Hamming(7,4) code at the expense of a longer transmission time. Since the further increase in transmission time is not desirable, a better approach to improve the accuracy is to find the threshold adaptively. Also, we find that the transmission is more accurate when the communication medium object is stable and there is less disturbance. Hence, the transmission accuracy can be further improved, if the transmitting device opportunistically transmits signals when the communication medium object is stable.

	Number of Correct Characters	**Accuracy(%)**
A	290	96.67
B	272	90.67
C	287	95.67
D	294	98.00
E	283	94.33
F	289	96.33

TABLE I

Accuracy of string transmission on six different communication medium objects.

* **A**: Stable wooden table, **B**: Wooden table with a user using computer, **C**: Metal shelf, **D**: Plastic shelf, **E**: Cushioned chair, **F**: Stack of copy paper.

B. Detection Threshold

We also conducted experiments by varying the threshold α. When the threshold is low, it is easier to detect a vibration signal on the surface but it is more sensitive to disturbances. On the other hand, when the threshold is high, it is more robust against disturbances but it may miss actual transmitted signal. Hence, finding the right value for the communication medium object and the surrounding disturbance level is the key to our proposed method. The experiment results from varying threshold value is shown in Figure 8. The experiment was conducted on a long narrow wooden table and the distance between a transmitter and a receiver was set to $50cm$ as before.

As shown in Figure 8, the highest accuracy appeared when $\alpha = 0.01$ (accuracy of 100%). This does not mean that 0.01 is the best threshold value but it is the best threshold value for the communication medium used in this experiment. When the threshold is lower than 0.0035, the receiver becomes too sensitive and detects movements when the transmitter has not transmitted a vibration. On the other hand, when the threshold is larger than 0.0175, the receiving device becomes too insensitive and even missed beacon bits. The accuracy increases up to $\alpha = 0.01$ and decreases afterwards. It shows

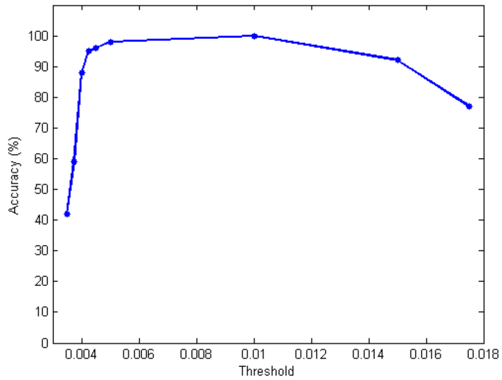


Fig. 8. The accuracy as a function of threshold α.

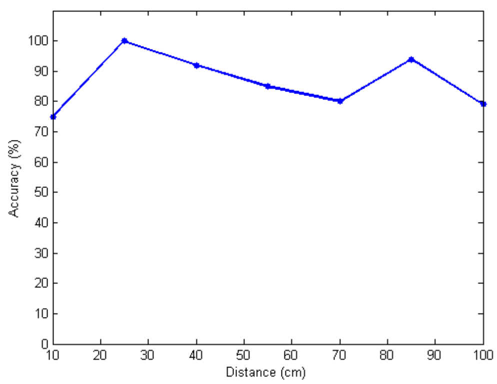


Fig. 9. The accuracy as a function of the distance between a transmitter and a receiver.

that there is an optimal value for α and we plan to adaptively find the optimal threshold in our future work.

C. Communication Distance

In this section, we tested our method by varying the distance between two devices. We set the threshold α to 0.015. As shown in Figure 9, when the distance between two devices is $25cm$, the accuracy is the highest (100%) and decreases as the distance increases. A surprising finding is that the accuracy is low when the separation between two devices is $10cm$ (the shortest distance we tried in this experiment). One reason for this can be the interference from the receiving device. When two devices are placed too close, the receiving device can change the vibrational pattern of propagation. In this case, the receiving device acts as a barrier against the propagation of vibration and generates some reflected vibration. The next interesting fact is that the accuracy is high when the distance is $85cm$. We believe that this is due to the configuration of the table used in the experiment and reflected vibrations might have been positively correlated with the original vibration signal, improving the magnitude of transmitted vibration signal.

VI. Limitations and Future Work

Current limitations of the proposed method are its long communication time and need for assigning the threshold for different conditions. The communication time can be reduced if a vibrating device with a higher frequency is embedded into smartphones. We believe that once vibration is adapted as another way to communicate using smartphones, the smartphone manufacturers will equip their smartphones with more sophisticated vibrators. Hence, we believe this problem can be easily solved. A more difficult issue is the assignment of the threshold value. As shown in experiments, there is no single threshold value that works the best for all cases. An ideal approach is to detect the communication medium device and the surrounding disturbance level and adjust the threshold autonomously. We plan to address this problem in our future work.

VII. Conclusion

This paper introduces a novel communication method without using radios. It utilizes the built-in vibrator and accelerometer of a smartphone which are currently available to all smartphones. We have implemented a system that can send and receive text information using vibration. The proposed approach allows communication among smart devices with a higher level of privacy at a longer distance compared to the currently available short-range communication methods. A prototype system has shown an accuracy of 95%, showing the feasibility of the proposed method.

VIII. Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. 2010-0027155).

References

[1] Wifi (wireless fidelity): http://www.wi-fi.org.

[2] P. Baronti, P. Pillai, V. W. Chook, S. Chessa, A. Gotta, and Y. F. Hu. Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards. *Computer Communications*, 30(7):1655 – 1695, 2007.

[3] K. Curran, A. Millar, and C. M. Garvey. Near field communication. *International Journal of Electrical and Computer Engineering (IJECE)*, 2, 2012.

[4] J. Haartsen. The bluetooth radio system. *Personal Communications, IEEE*, 7(1):28 –36, 2000.

[5] T. Komine and M. Nakagawa. Fundamental analysis for visible-light communication system using LED lights. *Consumer Electronics, IEEE Transactions on*, 50(1):100 – 107, 2004.

[6] H. Lu, J. Yang, Z. Liu, N. D. Lane, T. Choudhury, and A. T. Campbell. The jigsaw continuous sensing engine for mobile phone applications. In *Proc. Intl. Conf. on Embedded Networked Sersor Systems (SenSys) (2010)*, pages 71–84, 2010.

[7] R. Want. An introduction to RFID technology. *Pervasive Computing, IEEE*, 5(1):25 – 33, 2006.

[8] S. Weis, S. Sarma, R. Rivest, and D. Engels. Security and privacy aspects of low-cost radio frequency identification systems. In *Security in Pervasive Computing*, volume 2802 of *Lecture Notes in Computer Science*, pages 50–59. 2004.

[9] T. Yonezawa, T. Ito, and H. Tokuda. Transferring information from mobile devices to personal computers by using vibration and accelerometer. In *Proceedings of the 13th international conference on Ubiquitous computing*, UbiComp '11, pages 487–488. ACM, 2011.

[10] T. Yonezawa, H. Nakahara, and H. Tokuda. Vib-connect: A device collaboration interface using vibration. In *Embedded and Real-Time Computing Systems and Applications (RTCSA), 2011 IEEE 17th International Conference on*, volume 1, pages 121 –125, 2011.