AirLink: Sharing Files Between Multiple Devices Using In-Air Gestures

Ke-Yu Chen[1,2], Daniel Ashbrook[2], Mayank Goel[1], Sung-Hyuck Lee[2], Shwetak Patel[1]

[1]University of Washington, DUB, UbiComp Lab
Seattle, WA, USA
{kychen, mayankg, shwetak}@uw.edu

[2]Samsung Information System America
San Jose, CA, USA
{d.ashbrook,sung1.lee}@sisa.samsung.com

ABSTRACT
We introduce AirLink, a novel technique for sharing files between multiple devices. By waving a hand from one device towards another, users can directly transfer files between them. The system utilizes the devices' built-in speakers and microphones to enable easy file sharing between phones, tablets and laptops. We evaluate our system in an 11-participant study with 96.8% accuracy, showing the feasibility of using AirLink in a multiple-device environment. We also implemented a real-time system and demonstrate the capability of AirLink in various applications.

Author Keywords
Mobile phones; sensing; gestures; Doppler effect; multiple-device environment

ACM Classification Keywords
H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

INTRODUCTION
Sharing photos, text, and other information is a common task people engage in with their mobile devices. While many mechanisms for transferring this information exist, such as Bluetooth and WiFi, the ability for the user to select which device or devices to share with is still an open question. People still seem to resort to emailing content, even when the target device is just inches away. Existing interfaces such as those associated with Bluetooth or Apple's AirDrop require the user to browse through a possibly-lengthy list of receiving devices or users and choose where the information should be sent. Often the identifiers in these lists are ambiguous or simply the default name, making locating and selecting the desired device difficult. To work efficiently in a multiple-device environment (MDE), users need effective interfaces and communication channels among devices [2].

In this work, we introduce *AirLink*, a system that allows users to share files between multiple devices using in-air gestures (Figure 1). By waving the hand from one device to another, users can easily exchange information such as photos between multiple devices. AirLink leverages acoustic sensing to identify hand gestures. In particular, our algorithm measures the Doppler shift caused by hand movements and identifies the direction of hand movement from one device to another. By starting and stopping at the intended devices, there is no ambiguity as to which devices are sending and receiving the content. Furthermore, AirLink requires only standard microphones and speakers which are built into nearly all mobile devices.

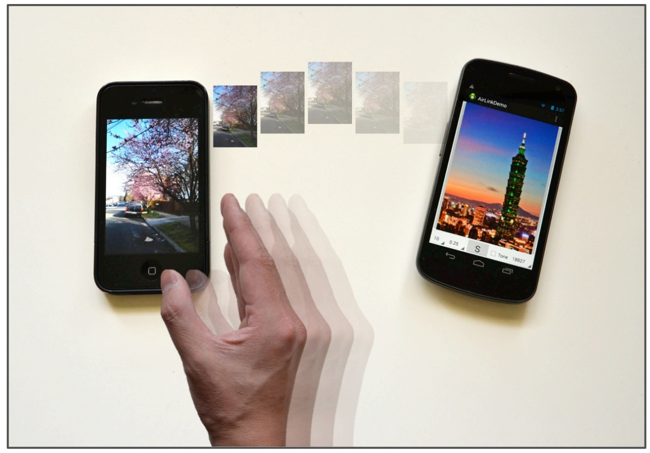

Figure 1: AirLink allows file sharing between multiple devices by using in-air gestures and without adding new sensors.

RELATED WORK
In-air gesture recognition using the Doppler effect was first demonstrated in SoundWave [4]. AirLink is similar in spirit to SoundWave; however the goal and algorithms are quite different. SoundWave was designed for detecting gestures on single device and was only evaluated on laptops. The feasibility of gesture detections using Doppler shift on mobile phones is still unknown. AirLink, on the other hand, was designed for multiple-device interactions. Our system combines signals from *all* devices and encodes them into a codeword which is used to identify various hand gestures (Figure 3). Besides point-to-point transmission, AirLink can broadcast a file to multiple nearby devices.

Researchers have explored various approaches to facilitate information sharing in MDE. Lucero et al. used a multi-emitter to track 3D positions of mobile phones and allow

UbiComp '14, September 13 - 17, 2014, Seattle, WA, USA
Copyright 2014 ACM 978-1-4503-2968-2/14/09…$15.00.
http://dx.doi.org/10.1145/2632048.2632090

users to share photos between phones [9]. Hardy et al. utilized a mesh of NFC tags and built-in RFID readers to enable 2D positioning and image transfer between phones [6]. Other techniques, based on computer vision, associated the phone with an interactive surface [13] or its user [11], allowing photos to be exchanged between devices. These approaches, however, require customized hardware or infrastructure, limiting their ubiquity. AirLink, on the other hand, uses standard microphones and speakers built into most mobile devices, obviating the need for extra hardware.

Like AirLink, other research has leveraged simple sensors to detect phone gestures and enable file sharing between devices. Yatani et al. used inertial sensors to detect a toss or swing gesture [14] and Hinckley utilized accelerometers to enable file sharing by bumping two devices together [7]. By detecting the Doppler shift caused by phone movements, researchers enabled device pairing by swinging the phone toward the target device [1,12]. Although gesturing with the device itself has been shown a promising technique in multiple-device communication [8], it may not scale well to all types of mobile devices: for example, laptops. Ogata et al. [10] used a custom-made stylus to connect devices; however, it required two hands to use (i.e., one hand holding the phone with the other holding the stylus). Goel et al. utilized surface-mounted piezoelectric sensors to detect hand gestures performed on the surface and to enable file sharing between devices resting on that surface [5]. In AirLink, users simply perform in-air hand gestures to trigger such file transfers, even for devices held in the hand.

THEORY OF OPERATION

AirLink leverages the Doppler effect to detect the direction of hand movements. The phenomenon of the Doppler effect is the shift in frequency of a signal caused by a moving object [3]. In AirLink, each device generates an ultrasonic pilot tone from its speakers. When the user wants to share a file, he/she simply waves the hand from the initiating device to the receiving device. The hand movement reflects the ultrasound, causing a shift in frequency. This change in frequency is captured by the phones' microphones, allowing gesture detection. The frequency shift perceived by a particular device can be expressed as

$$f = \left(1 + \frac{\Delta v}{c}\right) f_o \quad (1)$$

where f_o and f are respectively the emitted and perceived frequency of ultrasound on the phone, c is the speed of sound in air, and Δv is the velocity of hand movement relative to the device.

Figure 2 shows an example of the shifted frequency of a left-to-right gesture (illustrated in Figure 3 as "L2R"). With the pilot ultrasonic tone emitted at 18.8 KHz, the shifted frequency varies in response to the direction of hand movement. The hand moves away from device A (Figure 2 left), causing the pilot frequency to shift lower than its original value. The hand moves towards device C (Figure 2 right), causing it to detect a shifted frequency higher than the pilot tone. Device B, in the center, (Figure 2 middle) perceives the hand moving towards and then away, and therefore the frequency is detected shifting from higher than the pilot frequency to lower.

System and Implementation Details

AirLink requires a pilot tone continuously played through the device's speakers. Most speakers on phones and laptops are able to generate audio up to 22 KHz. In this work we selected the pilot tone at 18.8 KHz because it is usually inaudible for human while detectable by almost all standard microphones. With this ultrasonic tone being emitted, motions near the device cause Doppler-shifted reflections, which can be picked up by the microphones.

The system samples at 44.1 KHz, processing the signal in segments of 4096 samples (about 0.093s) at a time. For each segment, it computes the Fast Fourier Transform (FFT), yielding a 2048-point FFT vector with 10.77 Hz bins. The system then removes incidental noise by setting elements in the vector with values under 2.5% of the magnitude of the pilot tone to zero. This threshold value was chosen empirically (see Figure 4 and *Evaluation and Results* section). The result of this process is a stream of FFT vectors, which AirLink uses to recognize gestures.

Feature Extraction

The system uses a four vector-long (0.37s) moving window on the FFT stream; long enough for gesture recognition while avoiding significant delays in response time. AirLink extracts features from this window by counting the number of non-zero elements in each vector, incrementing the count for values higher than the pilot tone and decrementing for lower values. A higher count indicates a larger shift in frequency. To avoid false positives, the system counts only elements above or below10 bins (or 107.7 Hz) from the pilot frequency. By Equation 1, this restriction allows only hand movements faster than 1.95 m/s; as most people can move their hands at up to 3.9 m/sec [4], this 10-bin threshold is sufficient to detect hand gestures while eliminating slow hand movements or minor motions such as finger flicks. This procedure generates four numbers, representing the amount of frequency shift for each vector in the window. Finally, the counts are summed for the first two vectors and

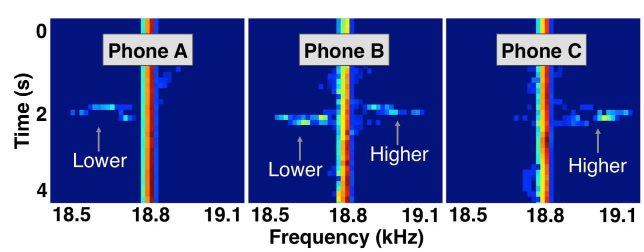

Figure 2. The detected Doppler-shifted signals of a L2R gesture (illustrated in Figure 3). This waterfall plot shows x-axis as frequency and y-axis as time. The pilot tone is emitted at 18.8 KHz on all three phones. On phone B (middle), the shifted frequency first appears on the *right* (higher than the pilot tone) and changes to the *left* (lower than the pilot tone).

the latter two vectors, yielding a feature vector $F = [F_1, F_2]$, which is used for gesture detection.

Hand Gesture Detection

Using simple calculations, AirLink is able to recognize three basic hand motions: hand moving *away* from the phone $(F_1 + F_2 < 0)$; hand moving *towards* the phone $(F_1 + F_2 > 0)$; and hand moving *towards and then away from* the phone $(F_1 > 0, F_2 < 0)$. This process takes place on multiple phones; when each phone detects a gesture, it sends the recognition result to a central server, which collates the time stamped gestures from all phones in order to enable sharing.

Decision Making

By combining the hand motions reported from multiple phones, the system is able to recognize the source, destination, and direction of a hand movement. Representing a movement *towards* a phone as T, *away* as A, and *towards then away* as X, a particular motion can be seen as a unique sequence of characters, or codeword. Figure 3 illustrates the gestures and accompanying codewords possible with three phones. For example, a L2R gesture is represented as *A-X-T*. Note that the actual relative positions of the devices does not need to be known *a priori*, but can be deduced from the reporting times of the characters in the codeword.

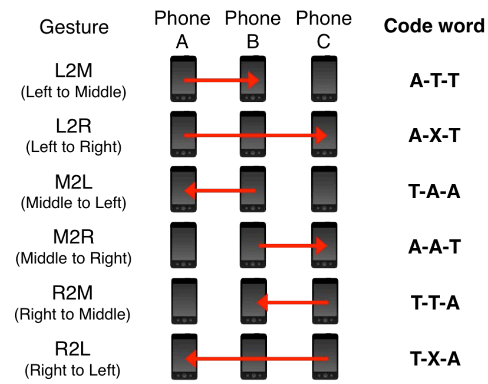

Figure 3. By combining 3 basic hand movements: *T* (toward), *A* (away), and *X* (toward and then away), each type of gestures can be represented as a unique codeword.

Broadcast

Besides the basic gestures for device-to-device transfer, AirLink also supports broadcasting from one device to multiple nearby devices. When a user wants to broadcast a file, she simply performs a "patting" gesture above the originating device. To recognize this motion, the system uses features from the four shift counts described earlier instead of using the two-tuple feature vector. In particular, if a device observes switching signs in the shift counts (i.e., "$+-+-$"), it recognizes this pattern as a patting gesture and sends the detection result to the server. Although all devices near the initiating device may detect the same tapping gesture, the device nearest the hand will cause the highest frequency shift. The server then groups all devices that report a patting gesture and chooses the device with largest number as the initiating device.

Relative Positions

AirLink is capable of deducing the relative locations of devices when a swipe gesture is performed (e.g., the L2R or R2L gesture in Figure 3). Because the Doppler-shifted signals occur sequentially according to the phone topology, the system can differentiate their relative positions. In a 3-phone scenario (Figure 2), the devices on both ends will detect a T (toward) or A (away) hand movement, while the device between them will detect an X (towards then away) at timestamps based upon their relative position.

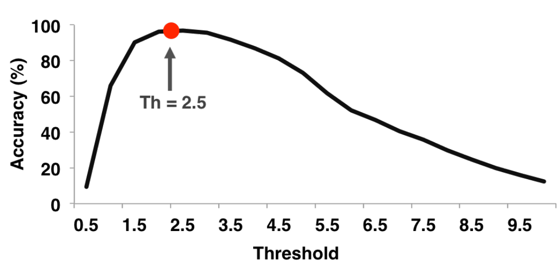

Figure 4. Threshold selection. The best threshold occurs at 2.5, yielding the average accuracy of 96.8% across 11 participants.

EVALUATION AND RESULTS

We set up our experiments in a laboratory environment and conducted a user study with 11 participants (4 female) in a 3-phone scenario. Three phones were placed in an almost straight line and 25cm apart.

Noise Removal Threshold Selection

In order to minimize false positives while maintaining sufficient accuracy, we chose the threshold for removing incidental noise by empirical evidence from the study data. The threshold is defined as a percentage of the magnitude of the pilot tone; any signals below the threshold will be regarded as noise and set to zero. Only those hand movements that cause sufficiently large Doppler-shifted reflections will be used by the system. Figure 4 shows the accuracy curve and indicates that the best threshold is at 2.5%.

Experiment I: Robustness and Results

Our first experiment tested the robustness of the AirLink system. We tested the six hand gestures shown in Figure 3: *L2M* (left-to-middle), *L2R* (left-to-right), *M2L* (middle-to-left), *M2R* (middle-to-right), *R2M* (right-to-middle) and *R2L* (right-to-left). These six gestures cover all possible hand movements a user can perform with three devices. Each participant performed ten repetitions of each gesture, with a total of 660 gestures performed in the study. We randomized the order of gestures to mitigate any temporal bias. Prior to the study, participants were told the appropriate speed of hand movements and practiced each of the six hand gestures for 3–5 minutes. During the study, participants were allowed to switch between their right and left hands. For consistency, we chose three identical phones (Samsung Galaxy Nexus) for this controlled study; however, we also show the feasibility of using the AirLink system on another phone (Samsung Galaxy S3) in the video figure.

Table 1 lists the accuracy of the six hand gestures. Overall, we found that AirLink can reliably differentiate these gestures in this three-phone scenario. The average accuracy (96.8%) shows the stability and feasibility of using AirLink in a multiple device environment. As we expected, the accuracy of the short-range hand gestures (i.e., M2L, M2R, L2M and R2M) is high. In these gestures, hand movements are usually performed stably with sufficient speed to trigger observable Doppler-shifted signals. On the other hand, longer-range hand gestures (i.e., L2R and R2L) resulted in a slightly lower accuracy (92.7% and 93.6%). We noticed that when some users performed these gestures, they did not move their hand far enough past the middle phone. The middle phone therefore identifies a *T* gesture instead of an *X* gesture, which degraded the overall accuracy. To further test the robustness of AirLink against noise, we placed the phones in a noisy restaurant for one hour without performing any gestures. During this period, we did not observe any false alarms. This finding is in accordance with[4]: the system uses ultrasound at a frequency distinct from surrounding sounds and is therefore robust in a dynamically changing environment.

Experiment II: Phone Topology and Results

We conducted a second study to test the AirLink system under different phone topologies, to evaluate how well auser can select one of two phones when those two phones are not located on a straight line. We chose a subset of participants (2 males, 1 female) to perform four gestures (i.e., M2L, L2M, M2R and R2M). Each participant performed ten repetitions of each of these four gestures. In this experiment, we kept the same distance between the initiating phone and the receiving phone (25cm) while changing the angles between two receivers (see Figure 5). As shown in Table 2, the accuracy degrades as the angle between receivers decreases. This is because when two phones are closer, they perceive more similar Doppler-shifted signals (e.g., both phones perceiving the hand moving toward them). This experiment shows that AirLink performs reliably for angles from180° to 120°.

DISCUSSION AND FUTURE WORK

Although our study focused on the robustness of AirLink in a 3-phone scenario, our approach can in theory be extended to more phones. For example, in a 5-phone setup (say, phone 1 to 5 from left to right) where the user waves their hand from phone 1 to 4, the corresponding codeword is AXXTT. In this case, the confusion occurs between phone 4 and 5 since they would both detect a T gesture. Phone 4, however, would perceive a higher Doppler shift than phone 5 since it is closer to the hand. Therefore, it could infer that the gesture ends at phone 4 rather than phone 5. We plan to test this scenario and extend support to more than 3 phones.

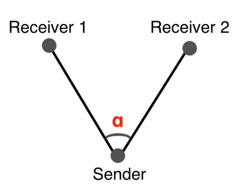

Figure 5. Topology of sender and receiving phones in Experiment II.

α	Accuracy (%)
180°	98.6
150°	93.5
120°	89.2
90°	80.0
60°	70.0

Table 2. Average % of correctly recognized hand gestures between two receivers under different angles α.

Hand Gesture		Accuracy (%)
M2L	(Middle-to-Left)	100
M2R	(Middle-to-Right)	97.3
L2M	(Left-to-Middle)	99.1
L2R	(Left-to-Right)	93.6
R2M	(Right-to-Middle)	98.2
R2L	(Right-to-Left)	92.7

Table 1. Average % of correctly recognized hand gestures across 11 participants.

Besides file sharing, AirLink could enable *device pairing* such as [1,5], adding another dimension of interaction. To do so, the user performs an initiating gesture to connect the devices by simply waving the hand between them. After two devices are linked, users can sync photos between them (similar to Apple's AirPlay) or play games.

Continuously generating ultrasonic tones could drain the battery of the cell phone very quickly and may trigger undesired events due to nearby hand movements. To mitigate these issues, the device can be configured to emit ultrasonic tones only when specific applications are being used (e.g., photo browser or email). This tone triggering process could even become automatic by analyzing user behaviors. We leave this functionality to future work, however.

In our current implementation, all devices play the same frequency tone. AirLink's capabilities can be further extended by playing a different frequency on each device. By doing so, it may be possible, without the need to perform a swipe gesture, to determine the phones' relative positions by triangulation. Furthermore, multiple tones could be used to detect the proximity of a newly joined device. For a stationary device such as a TV, a tone at a specific frequency can be continuously played as a heartbeat. When a phone enters the physical space and detects the heartbeat, it can automatically register with the local server for interaction.

CONCLUSION

In this work, we demonstrated an approach for easy file sharing between multiple devices using in-air gestures. Our approach leverages the Doppler effect caused by hand movements to enable both device-to-device transfer and one-to-all broadcasting. The system differentiates hand gestures by combining the recognition results reported from multiple devices; the combined hand movements together form a unique codeword, representing one of the possible hand gestures between those devices. We showed the feasibility of AirLink through an 11-user study and a real-time implementation. Our approach requires only standard speakers and microphones, which can be found on nearly all mobile devices, enabling it to be implemented in a multiple-device, cross-platform environment.

REFERENCES

1. Aumi, M. T. I., Gupta, S., Goel, M., Larson, E., and Patel, S., DopLink: using the Doppler effect for multi-device interaction. In *Proc. UbiComp 2013*, pp.583–586.
2. Biehl, J. T., and Bailey, B. P. Improving interfaces for managing applications in multiple-device environments. In *Proc. AVI 2006*, pp.35-42.
3. Doppler, Christian. Über das farbige, Licht der Doppelsterne und einigeranidererGestirne des Himmels. In *Proceedings of the Royal Bohemian Society of Sciences (Part V, Vol 2)*, Prague, 1842.
4. Gupta, S., Morris, D., Patel, S., and Tan, D., SoundWave: Using the Doppler Effect to Sense Gestures. In *Proc. CHI 2012*, pp.1911-1914.
5. Goel, M., Lee, B., Aumi, M.T., Patel, S.N., Borriello, G., Hibino, S., Begole, J., SurfaceLink: Using Inertial and Acoustic Sensing to Enable Multi-Device Interaction on a Surface. *In Proc. CHI 2014*, pp.1387-1396
6. Hardy, R., and Rukzio, E., Touch & interact: Touch-based Interaction of Mobile Phones with Displays. In *Proc.MobileHCI 2008*, pp.245-254.
7. Hinckley, K., Synchronous gestures for multiple persons and computers. In *Proc.UIST 2003*, pp.149-158.
8. Kray, C., Nesbitt, D., Dawson, J., and Rohs, M., User-defined gestures for connecting mobile phones, public displays, and tabletops. In *Proc. MobileHCI 2010*, pp.239-248.
9. Lucero, A., Holopainen, J., and Jokela, T., Pass-them-around: collaborative use of mobile phones for photo sharing. In *Proc. CHI 2011*, pp.1787–1796.
10. Ogata, M., Sugiura, Y., Osawa, H., and Imai, M., Flash-Touch: Data Communication through Touchscreens. In *Proc. CHI 2013*, pp.2321-2324.
11. Rofouei, M., Wilson, A., Brush, A. J., and Tansley, S., Your phone or mine? Fusing Body, Touch and Device Sensing for Multi-User Device-Display Interaction. In *Proc. CHI 2012*, pp.1915-1918.
12. Sun, Z.,Purohit, A., Bose, R., and Zhang, P., Spartacus: spatially-aware interaction for mobile devices through energy-efficient audio sensing. In *Proc. MobiSys 2013*, pp.263-276
13. Wilson, A. D., and Sarin, R., BlueTable: Connecting Wireless Mobile Devices on Interactive Surfaces Using Vision-Based Handshaking. In *Proc.GI 2007*, pp.119-125.
14. Yatani, K., Tamura, K., Hiroki, K., Sugimoto, M., and Hashizume, H., Toss-It: Intuitive Information Transfer Techniques for Mobile Devices. In *Proc.CHI EA 2005*, pp.1881–1884.