A Real Time Wireless Interactive Multimedia System

Hong Li, Wei Yang(✉), Yang Xu, Jianxin Wang, and Liusheng Huang

University of Science and Technology of China, Hefei, China
qubit@ustc.edu.cn

Abstract. Recent years, various interactive multimedia systems have been applied to relevant fields such as education, entertainment, etc. Researchers exploit sensors, computer vision, ultrasonic, and electromagnetic radiation to achieve human-computer interaction (HCI). This paper proposes an interactive wireless multimedia system which utilizes ubiquitous wireless signals to identify human motions around smart WiFi devices. Compared with related work, our system realizes interactions between human and computer without extra hardware devices. The system identifies human gestures around the smart devices (i.e., a laptop) equipped with the commercial 802.11n NIC, and it maps different gestures into distinguishable computer instructions. We build a proof-of-concept prototype using off-the-shelf laptop and evaluate the system in a laboratory environment with standard WiFi access points. The results show that our system detects human gesture with an accuracy over 95 % and it achieves an average gesture classification accuracy of 89 % for five different users.

Keywords: Gesture recognition · Human-computer interaction · WiFi

1 Introduction

Recent years witness a rising trend to incorporate gesture recognition system into various smart devices, including smart phones [1], laptops [2], gaming console [3]. These systems generally exploit the available sensors to enhance their functionality. The existing solutions adopt techniques such as computer vision [3], sensors [4–6], ultrasonic [2], and infrared to realize gesture recognition. These technologies are promising, however, they face some unavoidable disadvantages, including sensitivity to lighting conditions, requiring specialized hardware devices.

Given that the disadvantage of above techniques, WiFi-based gesture recognition [7–9] systems have been proposed to overcome the limitations of existing gesture recognition systems. These solutions are able to recognize in-air without extra equipments such as sensors or cameras. WiFi-based gesture recognition systems are based on analysis of the characteristics of signal patterns, including rising edge, falling edge, plateaus, caused by human motions. However, these system need sophisticated hardware devices to extract the desired signal features.

F. Li et al. (Eds.): APWeb 2016, Part I, LNCS 9931, pp. 53–65, 2016.
DOI: 10.1007/978-3-319-45814-4_5

For example, WiSee [8] and WiVi [9] adopt Universal Software Radio Peripheral (USRP) and the device-free radio-based activity recognition (DFAR) scheme [7] utilizes Software Defined Radio (SDR). Moreover, all these systems do not provide fine-grained interactions with a certain application in smart devices.

This paper presents a new method for controlling multimedia systems in smart devices by recognizing a set of human gestures under wireless environment. Our system does not need additional sensors, is resilient to environmental changes, and achieves recognizing human gestures in real time. The key insight is to leverage the effect of in-air gestures toward fine-grained channel state information (CSI) to recognize users' gestures. After that, the system maps the identified human gestures into intended instructions in smart devices to achieve system control. There are several challenges which must be solved in order to translate the above high-level idea into a practical system, including handling the noisy CSI time series due to multipath reflections, extracting and recognizing human gestures in CSI values, dealing with the variations of gestures as well as their attributes for different humans and even for the same human at different time.

To address these challenges, our system adopts Butterworth low pass filter to reduce the high frequency noise exists in CSI time series. The frequency of variations caused by the movements of hands lie at the low end of the spectrum, however, the frequency of noise lies at the high end of the spectrum. Butterworth low pass filter is a natural choice for eliminating these high frequency noise because its high fidelity in preserving both time and frequency resolution of WiFi signals. To extract and recognize subtle changes caused by human gestures in CSI time series, we introduce a unique signal pattern, for example, a preamble, to identify the beginning of the human gestures and counter the interference from irrelevant people. This also helps to enhance system's energy-efficiency which stems from the fact that detecting the preamble can be easily done by monitoring a simple threshold, rendering the system idle most of the time.

In summary, we make the following contributions in this paper:

(1) We present a proof-of-concept prototype on off-the-shelf laptops which extracts the physical layer CSI from the Intel 5300 NIC using a modified driver developed by Halperin *et al.* [10] to recognize a group of basic in-air gestures. Further, we use the identified gesture to control multimedia system in the smart WiFi devices.
(2) To evaluate the performance of this non-intrusive and device-free scheme, we test our system in our laboratory environment which covers an area of $50 \times 23\,\mathrm{ft}^2$ with only one target user. The gesture set includes 7 gestures (6 normal gestures and 1 preamble gesture). Each gesture is performed by a target user for 30 times. Finally, we get 1050 gesture instances for 5 different users to evaluate our system. The experimental results show that our system can detect human gesture with an accuracy over 95 % using a single assess point within a distance of 1 ft around smart WiFi devices, and it achieves an average classification accuracy of 89 % for 5 different users in a multimedia player application case study.

2 Preliminary

Smart WiFi devices that support IEEE 802.11n/ac standards generally have multiple transceiver antennas. Hence, they support multiple-input multiple-output (MIMO) which provides several MIMO channels between transmit-receive (TX-RX) antenna pairs. Each TX-RX pair of transmitter and receiver consists of multiple subcarriers. These WiFi devices keep monitoring the MIMO channels to effectively acquire the signal strength, Signal to Noise Ratio (SNR), transmit power and rate adaptations. These devices quantify the detailed state of channel information in terms of channel state information (CSI). Recently, Halperin *et al.* proposed a new methods [10] to acquire fine-grained CSI values by modifying the commercial 802.11n NIC. It extracts the primitive signal variations from the physical layer. As the extracted signals is the resultant of constructive and destructive interference of multipath signal reflection. The variations caused by gestures are captured in the CSI time series for all subcarriers between every TX-RX antenna pair. Then the variations can be extracted to identify gestures. In frequency domain, the narrowband flat-fading channel with MIMO. A MIMO system at any time instant can be expressed as follows:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}, \tag{1}$$

where $\mathbf{y}$ is the received vector, $\mathbf{x}$ is the transmitted vector, $\mathbf{n}$ represents the noise vector and $\mathbf{H}$ denotes the channel matrix. CSI is an estimation of $\mathbf{H}$. In Orthogonal Frequency Division Multiplexing (OFDM) system, CSI is represented at subcarrier level. CSI values in a single subcarrier can be formulated in the following equation:

$$h = |h|e^{j \sin \theta} \tag{2}$$

where $|h|$ and θ are the amplitude and phase respectively. Compare to Received Signal Strength Indicator (RSSI), CSI comprises fine-grained information. Hence, CSI can be utilized to sense subtle changes caused by human gestures.

3 System Conceptual Overview

In this section, we give the conceptual overview of our system including the Signal Processing, Gesture Set, and Multimedia Application Instruction.

(1) Signal Processing: This layer detects and extracts primitive CSI values in CSI streams, which reflect the signal diversity and space. These signal changes include rising edge, falling edge, pause. They are separately caused by moving the hand away from the receiver, moving the hand towards the receiver, and holding the hand still over the receiver. Other complicated gestures can be composed by combining these three variances.

(2) Gesture Set: Different CSI waveform patterns extracted from the primitive signals can be exploited to recognize higher level gestures. For example, an up-down hand gesture can be mapped to the primitive rising edge and

then falling edge. We define a set of gestures which can be represented by some primitive falling edges, rising edges and pauses. Considering all up-down, right-left or other gestures may have the similar effect on the signal variations and hence the same primitive sequence of a rising and then falling edge. We empirically choose the most suitable ways and positions to perform gestures which fit the applications and can be easy to distinguish and identify.

(3) Multimedia Application Instruction: We map the identified human gestures into a group of application instructions in this layer. We assume that each kind of gesture corresponding to a specific application instruction. As an example, for a multimedia system, a "pause" action can be performed with a push hand gesture, while a "speed up" action can be mapped to a right-movement gesture. In the next section, we give the details of system flow of extracting these different semantics and the relevant challenges.

4 System Design

In this section, we present the detail flow of our system and address the mentioned challenges. Our system flow covers three main procedures which corresponding to the system conceptual overview: Primitive Signal Processing, Gesture Recognition, and Gesture Mapping.

4.1 Primitive Signal Processing

The CSI values extracted from commodity WiFi Network Interface Cards (NIC) are inherently noisy because of the frequency changes in internal transmission rate, transmit power levels and even unavoidable Carrier Frequency Offsets (CFO) resulted from the hardware imperfections and environment variations [11]. To detect and extract human gesture information from CSI values, we must remove these innate noise. We empirically employ weighted moving average method for every 60 points to smooth the original signals. And then, the algorithm removes the DC component that accounts for the static reflections of the environment by subtracting the average value of CSI within a window containing 30 CSI values. Considering the high-fidelity of Butterworth filter, we first adopt a Butterworth low pass filter to remove high frequency noise which prevents us to identify human gestures. As the gesture movements while instructing applications around smart devices lie anywhere between 1 to 60 Hz, and the CSI sample rate is $F_s = 500$ samples/s, we set the cut-off frequency of Butterworth low pass filter with $w_c = \frac{2\pi * f}{F_s} = \frac{2\pi * 60}{500} \approx 0.75$ rad/s. To better compare the filtered signals with threshold, the system maps the filtered CSI values between their maximum and minimum interval. As can be seen in Fig. 1, we present a Up-Down gesture waveform as an example after the process of signal processing procedures. After that we obtain the filtered CSI time series. Assume that t represents the number of transmitting antenna and r represents the number of receiving antenna, then, we get a CSI matrix $\mathbf{M}_{t,r}$ with a dimension of $N \times T$,

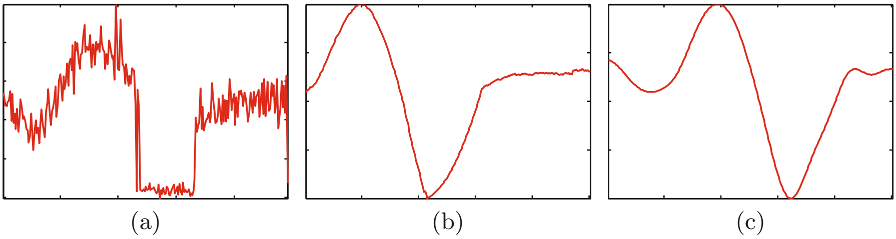


Fig. 1. Up-Down gesture waveform after different signal processing procedures. (a) Original signals (b) After applying weighted moving average method (c) Through low pass butterworth filter

where N is the number of CSI streams and T is the length of time. The value of N is related to the number of transmitting and receiving antennas. N can be calculated as $N = t \times r$ ($N = 9$ in our system), and we totally obtain $30 \times N$ CSI sucarriers. When a human gesture happens around our system, we experimentally observed that the CSI values change in all subcarriers. Hence, we find that the different subcarriers are correlated. In order to detect the human gestures in CSI time series, the system splits the CSI values of each subcarrier into R bins. We empirically set the bin size to be 100 CSI values to acquire the target CSI subcarrier which could be used for gesture detection. Then the algorithm calculates the variance of those bins. We compare the variances calculated for different bins of one subcarrier with the corresponding bins of other subcarrier, the subcarrier which has a larger number of higher variance bins is selected to be the target CSI subcarrier to detect gestures.

4.2 Gesture Recognition

After choosing the target CSI subcarrier. We use the target subcarrier to extract human gestures and their characteristics (i.e., frequency and waveform). It has the following two procedures: Detection and Recognition.

Detection: The gestures selected in our system are comprised of simple rising edges, falling edges, or pauses. To correctly detect the starting and finishing points of human gestures in target CSI subcarrier, we set thresholds to detect the occurrence of human gestures. The processed CSI values changes around the zero point, and the gesture waveforms lie both up and below the zero value. Hence, we set two thresholds to automatically detect human gestures. The positive threshold value is greater than zero which used for detect the gesture waveforms such as "Up". The negative threshold value is smaller than zero which facilitates the detection of the gesture waveform such as "Down". We empirically determined appropriate values of the two threshold. This method could efficiently detect the occurrence of target human gestures in real time. For the sake of saving energy and reducing the possibility of false detection. Our system sets a special preamble gesture as the commander to access the control right to the multimedia system. The preamble gesture is performed by user's waving hand twice towards the smart devices. It will lead to two regular convex peaks in the CSI waveform. After detecting the target gestures, the next stage is to search for two regular convex peaks, which indicates the preamble's happening. Once the preamble gesture is detected, the communication channel between the multimedia and the target user is built. And the system scans for various gestures according to the primitive CSI values. Otherwise, the system runs in a lower-power mode.

Recognition: Since different human gestures tend to cause different CSI changes in target CSI subcarrier, we can identify gestures by extracting CSI time series patterns caused by human gestures to achieve recognition. The system detects the onsets of target gestures by comparing CSI values with the defined thresholds. If the CSI value exceeds the value of the positive threshold or decreases to the value of negative value, it estimates the starting point of human gesture as s. We observe that on average the waveforms of a gesture spanned $t_{avg} = 500$ CSI values. Hence, we approximately get the finishing point as $e = s + t_{avg}$. Considering some gestures might have positive and negative waveforms such as "Up-Down". Then if the distance of two consecutive detected waveform less than d, the algorithm combines the two waveform to represent a same gesture. Finally, we set a guardian interval B which helps to extract the gesture waveforms. That means we add the guardian interval to both sides of the estimated gesture interval. Therefore, the gesture interval becomes $[s-B, e+B]$. Once the gesture onsets are determined, the algorithm extracts the CSI waveform between the gesture interval to identify gestures. We calculate the features from the acquired gesture waveform including zero-crossing rate, average value of gesture waveforms, first quartile and third quartile, variance, short time energy, short time average amplitude. We use the extracted features to form a feature vector to train FT, Naive Bayes (NB), and Random Forest Classifiers [12], respectively. We choose the classifier which has the best recognition performance to recognize gestures. Then the trained classifier can be used for recognizing the human gestures in real time.

4.3 Gesture Mapping

This section presents the direct mapping step based on the multimedia semantics. We map the application actions to their corresponding gestures as Table 1. After recognizing the human gestures using the pre-determined sequences, the system maps the identified gestures into their corresponding multimedia actions to control multimedia system. The gesture set in our interactive multimedia system generally covers 7 gestures which map to the most common 7 application actions in a multimedia system. Figure 2 shows the filtered waveforms of six gestures in our system. The developer can extend the gesture set and fully utilize the gesture attributes to enhance system's functionality. For example, the frequency attribute of gestures can be used to determine how fast the character should move in the multimedia system. We also note that multiple actions can

Table 1. Gestures and corresponding multimedia actions.

Human gestures	Multimedia actions
Up	Volume up
Down	Volume down
Up-Down	Play
Right	Speed up
Left	Slow down
Push	Stop

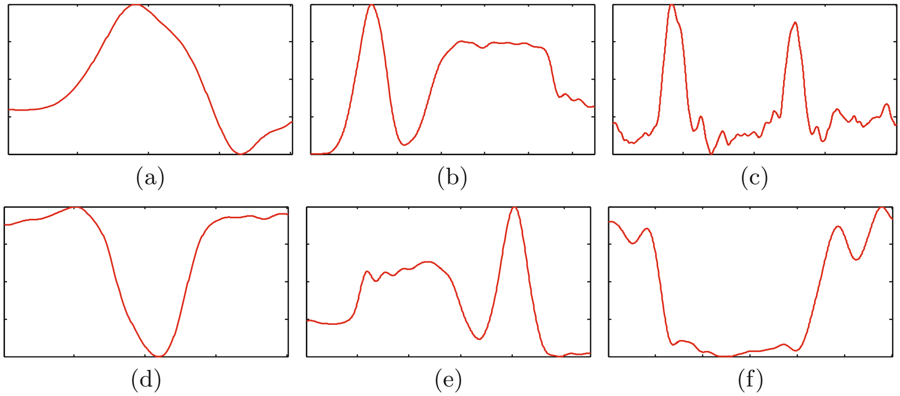


Fig. 2. The waveform of gestures in our system. (a) Up Gesture (b) Right Gesture (c) Wave Hand Twice (d) Down Gesture (e) Left Gesture (f) Push Gesture

be mapped to some other multimedia instructions such as the double right-hand could be mapping into speed up two times.

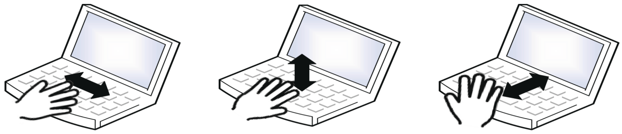
5 Evaluation

In this section, we analyze the performance of our proposed system in a typical laboratory environment. We first present the experimental setup in our environment and then we show the performance of our system.

5.1 Experimental Setup

The experimental setup includes two parts: hardware Setting and Data Collection. The details are illustrated below.

Hardware Setting. The system consists of two components: a laptop equipped with a commercial 802.11n WiFi card as a receiver and a commercially available WiFi access point (AP). We implement a proof-of-concept prototype of the



(a) Right-Left (b) Up-Down (c) Push

Fig. 3. The waveform of gestures in our system.

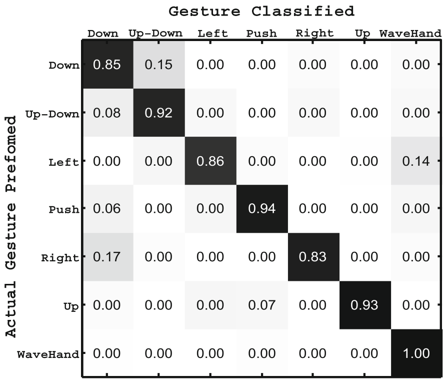


Fig. 4. Confusion matrix for the different gestures using NB classifier

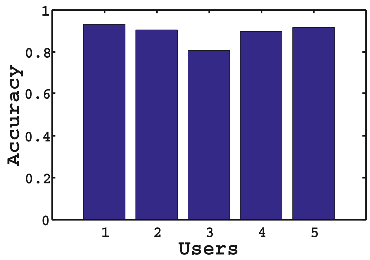


Fig. 5. Average gesture recognition accuracy using NB classifier

system in a Think-pad E40 laptop with Intel 5300 WiFi card and test it using a TP-LINK TL-WDR4300 wireless router as an AP. Both the receiver and the AP have 3 working antennas. The distance between the receiver and the AP is around 8 ft. To obtain CSI values from regular data frames transmitted by the AP, we modified the firmware of the WiFi card as in [10] to report CSI values to upper layers. All the experiments were performed in the 5 GHz frequency band with 20 MHz bandwidth channels. The system acquires CSI measurements from the CSI tool and processes it in real-time using MATLAB.

Data Collection. Our laboratory environment covers an area of $50 \times 23\,\text{ft}^2$. There is only one target user in the experimental environment. The target user performs gestures near the receiver with a distance about 1 ft. Figure 3 shows the movement of hand gesture near the receiver. We collect gesture dataset from five student volunteers also mentioned as users 1–5. Users 1–5 performs each gesture for 30 times. We totally collect 1050 gesture instances for performance evaluation of the system.

5.2 Performance Evaluation

In this section, we present the system performance in various conditions such as different classifiers, different users as well as different time during a daytime.

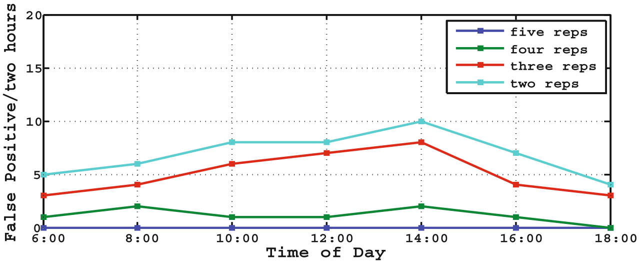


Fig. 6. False detection rate from a 12-h daytime trace

Different Classifiers. We test the performance of three classical classifiers (FT, NB, and Random Forest) in our experiment. We set feature vectors extracted from gesture instances of user 5 as the input samples for the three classifiers. These classifiers perform 10-fold cross validation. The result yields that the overall performance of the three classifiers are all above 90 %. However, the recognition accuracy of gesture "Right" in FT and Random Forest classifiers are 67 % and 50 %, respectively. In NB classifier, the recognition accuracy of all gestures are above 80 %. Figure 4 presents the confusion matrix for the seven gestures using NB classifier. Especially, the recognition accuracy of "WaveHand" reaches 100 %. It means our system can correctly identify the commander gestures of target user. Hence, we adopt NB classifier in our system for gesture recognition.

Different Users. To verify the system's resilience towards different users, the feature vectors of gesture instances collected from users 1–5 are used as the input of the selected NB classifier. We trained 5 user-specific NB classifiers for these 5 users. Every classifier performs 10-fold cross validation using each user's gesture instances. Figure 5 shows the average recognition accuracy of the gesture instances of these five users. Obviously, their average recognition accuracies are all above 80 %. The lowest extraction accuracy for user 3 shows that more gestures were falsely classified, which is due to the significant difference in his gesturing behavior compared to other users. The speed and magnitude of users' gestures also influence the recognition accuracy of our system. High gesturing speed will lead to short time span of gesture instance. And on the other hand, if users perform gestures in a larger magnitude, the amplitude of the signal change will be much greater than the original signal level. The accuracy of our system for such a user can be increased significantly by adjusting the thresholds of our algorithm for the given user.

Different Time. We test the robustness of our system during a daytime (6:00 AM to 6:00 PM) with a time span of 12 h. Figure 6 plots the number of false detection events every two hours as a function of time. The figure shows results for different number of repetitions in the preamble. The average number of false events is highest when the preamble contains only two repetitions. And with the number of repetitions increases, the false detection events significantly decline. Specifically, with four repetitions, the average false detection rate reduces to 0.67 events per hour. When the number of repetitions are more than four, the false detection rate is zero. This is reasonable because it is unlikely that typical human motions would produce five consecutive regularly convex CSI waveforms.

6 Related Work

Human-Computer Interaction (HCI) is the study about how computer technology influences human work and activities. These technologies generally cover from obvious computers to mobile phones, household appliances, car infotainment systems and even embedded sensors such as automatic lighting. Recent years, various techniques are used to HCI systems to improve user experience. The techniques include fundamental interaction styles such as direct manipulation, the mouse pointing device, and windows. Application types, like drawing, text editing, etc. And the Up-and-Coming Areas that will likely have the biggest impact on interfaces of the future, such as gesture recognition, multimedia, and 3D [13].

The fundamental interaction styles was first demonstrated by Ivan Sutherland in his PhD thesis about Sketchpad [14]. It enables the manipulation of objects using a lightpen, including grabbing objects, moving them, changing size, etc. Then the mouse pointing devices and other basic intersections were proposed by researchers. The application types of interactions, for example, the first drawing program presented by William Newmans Markup in 1975. Nowadays, researchers tend to integrate gesture recognition techniques to control devices. There are some products using the state-of-the-art techniques, for example, Xbox Kinect [15] adopts hybrid cameras to recognize human motions to realize human-computer interaction in multimedia systems. WiGest [16] extracts variations of the received signal strength indicator values to identify gestures.

Gesture recognition techniques have found a diverse set of applications, e.g., 3D in-air user-interface for mobile and laptops [17], remote control of home appliances [8], sterilized operation of medical devices and distraction-free management of in-car infotainment system [18,19]. The typical gesture recognition systems can be categorized into three types: computer vision based, sensors based, audio and radio based. Wahs *et al.* gave a comprehensive study in vision based techniques [18]. Recent arts include Xbox Kinect, LeaMotion [20] both utilize computer vision to recognize human gestures. Wearable or near-body sensing techniques such as Data glove, Sayre glove [21]. They use sensors in users' gloves to sense gestures of target users. Audio signals generated by mobile devices may be affected by human gestures. Researchers extract the resulting pattern to recognize human gestures [22,23], An alternative way extracts Doppler features from soundwaves reflected by human gestures relevant to interaction with computers [2]. WiSee [8] extends this approach to WiFi signals to identify 9 human gestures. After that, various WiFi based human motion work were proposed like WiVi [9], WiTrack [24], WiHear [25], Wikey [26], WiDraw [27], etc.

7 Conclusion

In this paper, we present a wireless interactive multimedia system that uses the fine-grained channel state information extracted from the physical layer to control the multimedia system by detecting and recognizing human gestures around a smart WiFi device. Our system does not need any extra hardware devices such as sensors, cameras, or sophisticated USRP platforms. We simply extract the CSI values by modifying the commercial 802.11n NIC. After applying typical signal processing methods, the system detects the variations in CSI time series caused by human gestures. Our system can realize interaction with multimedia systems in a smart WiFi device equipped with commercial 802.11n NIC (e.g., Intel 5300 NIC). We addressed the following system challenges including signal denoising, gesture extraction, interferences elimination. We evaluate the system in a typical laboratory environment using the gesture instances collected from 5 users. The results show that our system can accurately detect the target human gesture with an accuracy over 95 %, and it achieves recognize human gestures with an average accuracy of 89 % for five different users. This accuracy indicates that our system has the ability to use the ubiquitous wireless signals to sense human gestures to further control multimedia systems.

Acknowledgments. We would like to thank the anonymous reviewers for their valuable comments for improving the quality of the paper. This work was supported by the National Natural Science Foundation of China (No. 61572456) and the Natural Science Foundation of Jiangsu Province of China (No. BK20151241).

References

1. Toshibag55. http://www.engadget.com/2008/06/14/toshiba-qosmiog55-features spursengine-visual-gesture-controls/
2. Gupta, S., Morris, D., Patel, S., Tan, D.: Soundwave: using the doppler effect to sense gestures. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 1911–1914. ACM (2012)
3. Shotton, J., Sharp, T., Kipman, A., Fitzgibbon, A., Finocchio, M., Blake, A., Cook, M., Moore, R.: Real-time human pose recognition in parts from single depth images. Commun. ACM **56**(1), 116–124 (2013)
4. Cohn, G., Morris, D., Patel, S., Tan, D.: Humantenna: using the body as an antenna for real-time whole-body interaction. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 1901–1910. ACM (2012)
5. Harrison, C., Tan, D., Morris, D.: Skinput: appropriating the body as an input surface. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 453–462. ACM (2010)
6. Kim, D., Hilliges, O., Izadi, S., Butler, A.D., Chen, J., Oikonomidis, I., Olivier, P.: Digits: freehand 3d interactions anywhere using a wrist-worn gloveless sensor. In: Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology, pp. 167–176. ACM (2012)

7. Scholz, M., Sigg, S., Schmidtke, H.R., Beigl, M.: Challenges for device-free radio-based activity recognition. In: Proceedings of the 3rd Workshop on Context Systems, Design, Evaluation and Optimisation (CoSDEO 2011), in Conjunction with MobiQuitous, vol. 2011 (2011)
8. Pu, Q., Gupta, S., Gollakota, S., Patel, S.: Whole-home gesture recognition using wireless signals. In: Proceedings of the 19th Annual International Conference on Mobile Computing & Networking, pp. 27–38. ACM (2013)
9. Adib, F., Katabi, D.: See through walls with wifi!, vol. 43. ACM (2013)
10. Halperin, D., Hu, W., Sheth, A., Wetherall, D.: Tool release: gathering 802.11 n traces with channel state information. ACM SIGCOMM Comput. Commun. Rev. **41**(1), 53–53 (2011)
11. Wang, W., Liu, A.X., Shahzad, M., Ling, K., Lu, S.: Understanding and modeling of wifi signal based human activity recognition. In: Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, pp. 65–76. ACM (2015)
12. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. ACM SIGKDD Explorations Newslett. **11**(1), 10–18 (2009)
13. Myers, B.A.: A brief history of human-computer interaction technology. Interactions **5**(2), 44–54 (1998)
14. Sutherland, I.E.: Sketch pad a man-machine graphical communication system. In: Proceedings of the SHARE Design Automation Workshop, pp. 6–329. ACM (1964)
15. XboxKinect. http://www.xbox.com/en-US/kinect
16. Abdelnasser, H., Youssef, M., Harras, K.A.: Wigest: A ubiquitous wifi-based gesture recognition system. In: 2015 IEEE Conference on Computer Communications (INFOCOM), pp. 1472–1480. IEEE (2015)
17. Pickering, C.A., Burnham, K.J., Richardson, M.J.: A research study of hand gesture recognition technologies and applications for human vehicle interaction. In: 3rd Conference on Automotive Electronics. Citeseer (2007)
18. Wachs, J.P., Kölsch, M., Stern, H., Edan, Y.: Vision-based hand-gesture applications. Commun. ACM **54**(2), 60–71 (2011)
19. Melgarejo, P., Zhang, X., Ramanathan, P., Chu, D.: Leveraging directional antenna capabilities for fine-grained gesture recognition. In: Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pp. 541–551. ACM (2014)
20. LeapMotion. https://www.leapmotion.com/
21. Dipietro, L., Sabatini, A.M., Dario, P.: A survey of glove-based systems and their applications. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **38**(4), 461–482 (2008)
22. Tarzia, S.P., Dick, R.P., Dinda, P.A., Memik, G.: Sonar-based measurement of user presence and attention. In: Proceedings of the 11th International Conference on Ubiquitous Computing, pp. 89–92. ACM (2009)
23. Scholz, M., Riedel, T., Hock, M., Beigl, M.: Device-free and device-bound activity recognition using radio signal strength. In: Proceedings of the 4th Augmented Human International Conference, pp. 100–107. ACM (2013)
24. Adib, F., Kabelac, Z., Katabi, D., Miller, R.C.: 3d tracking via body radio reflections. In: 11th USENIX Symposium on Networked Systems Design and Implementation (NSDI 14), pp. 317–329 (2014)

25. Wang, G., Zou, Y., Zhou, Z., Wu, K., Ni, L.M.: We can hear you with wi-fi! In: Proceedings of the 20th Annual International Conference on Mobile Computing and Networking, pp. 593–604. ACM (2014)
26. Ali, K., Liu, A.X., Wang, W., Shahzad, M.: Keystroke recognition using wifi signals. In: Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, pp. 90–102. ACM (2015)
27. Sun, L., Sen, S., Koutsonikolas, D., Kim, K.H.: Widraw: enabling hands-free drawing in the air on commodity wifi devices. In: Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, pp. 77–89. ACM (2015)