AirThumb: Supporting Mid-air Thumb Gestures with Built-in Sensors on Commodity Smartphones

Tianrui Hu
School of Digital Media and Design Arts, Beijing University of Posts and Telecommunications
Beijing, China
School of Creative Media, City University of Hong Kong
Hong Kong, China
hutianrui@bupt.edu.cn

Taizhou Chen*
Department of Computer Science and Technology, Shantou University
Shantou, China
tzchen@stu.edu.cn

Kening Zhu*†
School of Creative Media, City University of Hong Kong
Hong Kong, China
keninzhu@cityu.edu.hk

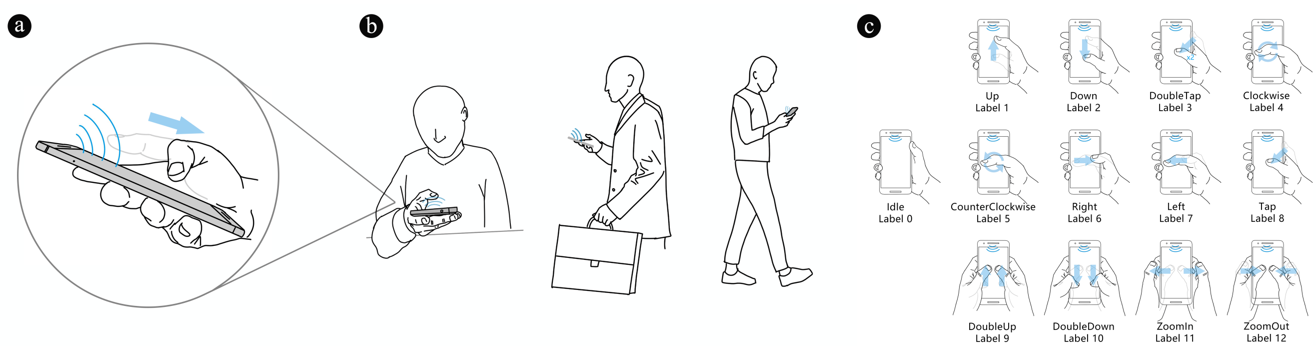

Figure 1: (a) AirThumb enable mid-air thumb-based interaction. (b) Usage scenarios of AirThumb, including: sitting, standing, and walking. (c) AirThumb recognizes 13 gestures, including one idle gesture (label 0), 8 one-thumb gestures (label 1-8), and 4 two-thumb gestures (label 9-12).

ABSTRACT

Taller and wider screens have become a new design tendency on current commercial smartphone market. However, the increasing size of the touch screen on the phone limits the interactivity of the user's thumb-based interaction. In this paper, we present AirThumb, a machine-learning-based sensing technique to support mid-air thumb gesture interaction on smartphones using the built-in sensors. AirThumb detects mid-air thumb-based gestures by leveraging multi-sensor data fusion technique, which combines the reflection pattern of an ultrasonic signal that propagates from the top speaker to the bottom microphone with subtle motion data from the phone's built-in IMU sensor. Our experiment shows that AirThumb achieves overall recognition accuracy of 94.55%, 94.52%, and 86.14% in sitting, standing, and walking scenarios, respectively. In addition, we demonstrate that the proposed multi-sensor data fusion technique enables AirThumb to quickly adapt to new users with fewer training samples required.

CCS CONCEPTS

• **Human-centered computing** → **Gestural input**; **Interaction devices**.

KEYWORDS

mid-air thumb gesture, sensor fusion, machine learning, mobile interaction

ACM Reference Format:
Tianrui Hu, Taizhou Chen, and Kening Zhu. 2025. AirThumb: Supporting Mid-air Thumb Gestures with Built-in Sensors on Commodity Smartphones. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA '25), April 26-May 1, 2025, Yokohama, Japan.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3706599.3721219

1 INTRODUCTION

Thumb-based interaction is often considered an efficient and convenient way to interact with mobile phones [14, 29], but its limitations, such as reduced dexterity and the increasing size of touch screens, hinder thumb-based smartphone tasks like typing [1], text editing [2, 21], and target selection [30]. Moreover, multi-touch gestures are difficult with one thumb [9], and the thumb can obstruct the

*Corresponding authors.
†Kening Zhu is also with CityU Shenzhen Research Institute.

CHI EA '25, April 26-May 1, 2025, Yokohama, Japan

ACM ISBN 979-8-4007-1395-8/2025/04
https://doi.org/10.1145/3706599.3721219

screen [2, 28]. A recent study also shows that users with long fingernails struggle with smartphone interactivity [7], and the experience worsens with wet or greasy fingers.

Various methods, including GUI design [15], on-screen thumb gestures [20], whole-device motion gestures [9], and back-of-device interaction [2] aim to improve one-handed interaction. One potential complement to standard on-screen touch input is using mid-air space above the screen for hand gestures [38]. This method allows users to interact without touching the screen, reducing finger obstruction and enabling interaction in situations where touch is not possible. However, current technologies require motions to be significant (e.g., hand swipes or circles) for recognition. Some existing systems for mid-air gestures require additional hardware, such as infrared motion capture [10], specialized touchscreens [13], or optical accessories [35]. Thus, there is a need for solutions that enable mid-air thumb gestures on low-cost, commercially available smartphones.

In this paper, we present AirThumb, a novel gestural interaction system that utilizes smartphone's built-in sensors, including the speaker-microphone pair and the inertial measurement unit (IMU), to recognize mid-air thumb gestures in various scenarios, such as sitting, standing, and walking (Fig.1 (a)(b)). AirThumb emits linear frequency-modulated ultrasonic chirp signals from the top speaker, which are affected by hand gestures during propagation. By learning the distortion in the received sound captured by the microphone, we can retrieve the thumb moving pattern in the near-surface above the screen. We also interoperate with the built-in IMU data to facilitate the gesture recognition task.

In this research, our main contributions are threefold:

(1) We developed AirThumb, a system for recognizing thirteen mid-air thumb-based gestures that leverage an acoustic sensing approach with the fusion of IMU sensor signals.
(2) We conducted a series of experiments to evaluate AirThumb . The results indicate that the AirThumb detects 13 mid-air thumb-based gestures accurately and robustly while users are sitting, standing, and walking. We also show that incorporating IMU sensor data improves gesture recognition accuracy, system robustness across scenarios, and system adaptiveness between users.
(3) AirThumb shows potential to expand the thumb-based input space from on the screen to above the screen, which unlocks the design space to support more intuitive and expressive interaction.

AirThumb offers a low-cost consumer-grade gesture recognition system that can be deployed on all commercially available mobile devices. This research shows great potential, contributing to enriching mobile interactions with novel interfaces. In our user study, participants expressed great interest in AirThumb and provided suggestions for its future investigation.

2 RELATED WORKS

2.1 Thumb-based Mobile Interaction

A major challenge faced in the context of mobile interaction is that devices are often used in a condition where the user has limited physical and attentional resources available [16]. For instance, some users frequently have their one hand occupied with holding or carrying objects, which significantly reduces the interactivity of mobile devices [27]. Given the nature of single-handed devices holding poses where the thumb is naturally placed above the screen [8], thumb-based interaction shows its privilege for single-handed mobile interaction [5, 15, 31, 32, 38]. Bergstrom-Lehtovirta et al. [3] have identified an optimal functional thumb-motion area for comfort and efficient thumb-based interaction, which inspired numerous researches on developing facilitating techniques for one-handed thumb interaction on mobile devices.

Although thumb-based interactions on the screen offer natural and efficient input experiences for mobile applications, challenges such as the fat-finger problem remain [30]. To this end, researchers have proposed using the space above the screen for interaction to support mid-air interaction [11, 12]. Kato Kunihiro and Ikematsu Kaori [17] supported above-screen thumb-posture detection through acoustic sensing. Their system enables in-air menu navigation by placing the thumb in different positions above the screen. Fabrice Matulic et al. [25] designed a two-mirror device to enable vision-based gestures recognition for VR context. Kunihiro and Lim [23] focused on detecting above-screen thumb-based posture through IMU-sensing rings. They recognized 20 gestures with an accuracy of up to 99%.

However, these approaches face limitations, often requiring external devices or modifications to the hardware, which hinders the widespread adoption of mid-air interaction. While Kato Kunihiro and Ikematsu Kaori [17] proposed works to support in-air thumb interaction without requiring external hardware, their method focused on detecting static thumb postures, which is still facing the limitation for function extension and usage scenario adaptation. AirThumb focuses on supporting mid-air thumb gesture-based interaction in multiple scenarios, such as sitting, standing, and walking.

2.2 Acoustic-based Gesture Sensing

Acoustic signal offers a non-invasive and contact-free sensing approach for localization or tracking, which has been widely used for many device-free gesture sensing tasks [4, 26, 33, 34, 36, 37]. One common approach to achieve this is using the Doppler Effect. AudioGest [34] enables mid-air gestures sensing on a laptop by transforming the device into an active sonar system that transmits inaudible acoustic signals from the built-in speaker. The system recognizes the gestures by decoding the distortion of the echoes using the Doppler shift. Similarly, Dolphin [33] and SonicOperator [22] utilized the Doppler Effect to support mid-air gesture interaction on the smartphone. Numerous research also detect mid-air gestures by modeling the acoustic signal's transition channel. UltraGesture [24] detected 12 desktop mid-air gestures at an accuracy of 97% by estimating the Channel Impulse Response (CIR). Strata [36] supported continuous finger tracking on the smartphone using similar sensing mechanism. Researchers also explore the use of the modulation technique commonly used in wireless communication to support fine-grained mid-air gesture detection. FingerIO [26] used Orthogonal Frequency Division Multiplexing (OFDM) to modulate the transmitted signal and tracked finger movements by analyzing signal distortion at the receiver, enabling centimeter-level continuous tracking. AirThumb is largely inspired by the aforementioned related works on acoustic-based mid-air sensing approaches.

We specifically focus on using chirp signals to recognize thumb gestures. Compared to previous studies, these gestures involve smaller motion ranges and are closer to the screen, making them more suitable for daily shortcuts and reducing the user's interaction burden. Furthermore, AirThumb has the potential to achieve better performance in dynamic thumb-based gesture detection.

3 USER PREFERENCE ON AIR THUMB INTERACTION

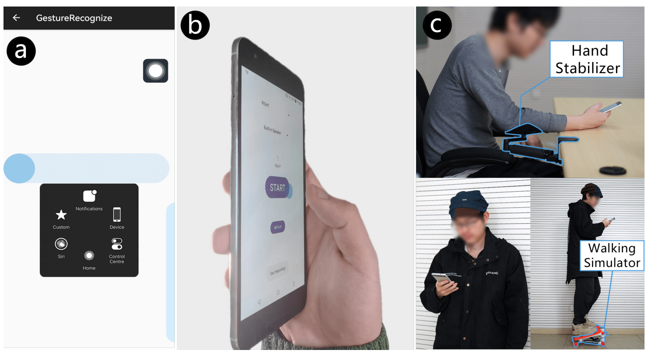

Figure 2: (a) Experienceable prototypes for the user study, including pop-up menu, edge gesture, and far-end buttons. (b) The data collection system interface. (c) Simulated sitting, standing, and walking scenarios for data collection.

Before implementing AirThumb, we investigated the user preference and acceptance of AirThumb interaction for mobile devices. A workshop study was conducted to compare the user preferences for three common mobile interaction interfaces: (A) Pop-up Menu (e.g. Apple's Assistive Touch); (B) Edge Gestures on Android (e.g., swiping from the screen edge to go back); (C) Far-end Buttons (e.g., the menu or back button in the top-left corner).

Participants tested all candidate interfaces on an Android phone, with experienceable prototypes for each (Fig. 2 (a)). An animated indicator bar simulated mid-air actions for AirThumb's preliminary interaction. Six participants from local universities were required to perform actions (e.g. 'go back,' 'go homepage') using different interfaces on our experienceable prototypes and ranked the interfaces by preference. Feedback was gathered through interviews. The study lasted about 30 minutes per participant.

As shown in Table 1, while far-end buttons were less favored, some participants attributed their lower ranking of AirThumb to habitual use of edge gestures (Android) or Assistive Touch (iPhone). Most participants (P1, P2, P5) reported no significant differences in fatigue between AirThumb and on-screen interactions, while P4 noted that AirThumb could reduce fatigue once users adapt. A participant with long fingernails (P5) found AirThumb less fatiguing than screen-based interactions. We summarized participants' feedback on mid-air thumb-based interaction in three aspects as follows, which provided potential design and implementation guidelines for our further exploration of AirThumb:

(1) AirThumb expands input channels: AirThumb could provide an additional interaction channel to reduce occlusion and simplify multi-touch actions, such as zooming and rotating, for one-handed use.
(2) AirThumb provides alternative interaction in specific scenarios: AirThumb enables seamless interaction when users' hands are wet, wearing gloves, or having long fingernails.
(3) Accidental trigger prevention: Users raise a concern about the false alarm by unintended near-screen finger movements.

4 AIRTHUMB DESIGN

4.1 Sensing System

4.1.1 Sensing Principle. AirThumb detects mid-air thumb movement by modeling the ultrasound transmission channel between the built-in speaker and the microphone of a smartphone. Specifically, we emit an ultrasonic signal from the top speaker, whose resonance characteristics would be changed if there is an obstacle (e.g. the thumb) moving on its propagation path to the bottom microphone. By learning the distortion in the received sound captured by the microphone, we can retrieve the thumb moving pattern in the near-surface above the screen. During practice, we also observed that the propagated sound would be distorted if the phone is non-static, namely, when the user is walking. We also observed that mid-air thumb movement would cause subtle movement of the smartphone. Therefore, we incorporate the built-in accelerator and gyroscope data from the smartphone as input features to facilitate the gesture classification task. The system structure is shown in Fig. 3.

4.1.2 Signal Processing. We adopted a similar sensing system pipeline as in previous research [18, 19], where we used the chirp signal as the baseband signal for the transmitter. The signal is a linear chirp divided into four segments, which is inaudible to users: 16–18kHz, 18–20kHz, 20–22kHz, and 22–24kHz, with each segment 200 samples long and 500 samples blank intervals between them. This structure helps minimize the impact of distant reflections. After being recorded by the mobile device's microphone, the raw signal passes through a high-pass filter (HPF) to remove low-frequency noise while retaining high-frequency features. A matched filter then identifies each segment, followed by FFT using a Hamming window with size 1024 to extract frequency features, with 171 discrete features selected from 16kHz to 24kHz. We sampled the accelerometer and gyroscope at 50Hz, resulting in a feature vector of length 6 per sample. During practice, we set each gesture to the length of 2 seconds. Therefore, we normalized each gesture recording to a signal clip consisting of:

$$2\text{ s} \div \left(\frac{200 \text{ samples} \times 4 + 500 \text{ samples} \times 4}{48000 \text{ Hz}} \right) \text{s} \approx 32 \text{ segments}$$

(where we dropped some of the initial signals). As a result, this process yields an acoustic feature vector of $\mathbb{R}^{171\times32}$ and an IMU feature vector of $\mathbb{R}^{6\times100}$ for each gesture.

4.1.3 Gesture Classification. We adopted a two-stage gating gesture classification strategy (Fig. 3) on the extracted feature as previous research suggested [6]. Specifically, we first trained a lightweighted binary classifier to perform gesture detection, namely, distinguish between *None* gestures and others. We then train a

Table 1: The table shows the workshop study results, users ranked four interaction methods by preference and comfort. AirThumb was consistently ranked 1st or 2nd, indicating strong user preference.

Participants	Rank 1st	Rank 2nd	Rank 3rd	Rank 4th
P1	Edge Gestures	AirThumb	Pop-up Menu	Far-end Buttons
P2	Edge Gestures	AirThumb	Pop-up Menu	Far-end Buttons
P3	AirThumb	Edge Gestures	Pop-up Menu	Far-end Buttons
P4	Pop-up Menu	AirThumb	Edge Gestures	Far-end Buttons
P5	AirThumb	Edge Gestures	Pop-up Menu	Far-end Buttons
P6	AirThumb	Pop-up Menu	Edge Gestures	Far-end Buttons

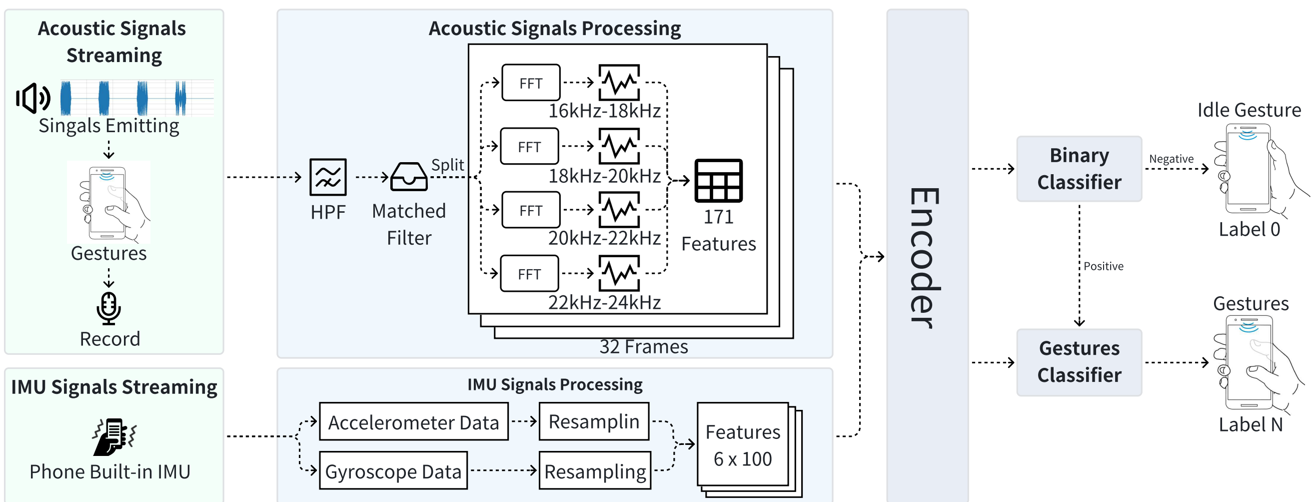

Figure 3: AirThumb system pipeline. AirThumb consists of three main stages, including multi-sensor data streaming, signal processing, and two-stage gesture recognition.

high-precision gesture classifier to achieve fine-grain classification results. We evaluated our method across a set of machine-learning classifiers in Section 5.

5 EVALUATION

5.1 Gesture Design and Scenarios

We evaluated AirThumb on a dataset consisting of 13 distinct mid-air thumb gestures, including 8 one-thumb gestures, 4 two-thumb gestures, and a default idle gesture, under three usage scenarios. To form a representative dataset of mid-air thumb movements, we summarized and refined commonly used gestures from previous studies[12, 38]. Specifically, we incorporated gestures such as directional thumb swiping, thumb cycling, and tapping in mid-air. Moreover, we included two-thumb gestures in our gestures set by expanding directional swiping from one thumb to two thumbs. The gestures are shown in Fig. 1 (c). In addition, we also performed gestures when users are in different situations, where we identify three typical smartphone usage scenarios: sitting, standing, and walking (Fig. 1 (b)). We also incorporate a *None* gesture as an idle gesture. As a result, we form a dataset consisting of 13 (gestures) × 3 (scenarios) distinct mid-air thumb gestures.

5.2 Data Collection

We implemented a data collection application on a Google Nexus 6P phone for the study. The application simultaneously records acoustic data and IMU data for every gesture and stores them locally. We recruited 13 participants (7 females, 6 males) for the data collection study. Their average age was 24, and 92.3% of them are right-handed users. Each participant was required to record all 13 gestures in three scenarios: sitting, standing, and walking, presented in a Latin-square counterbalanced order. To reduce fatigue, we provided a hand stabilizer for the sitting scenario and a walking simulator for the walking scenario (Fig. 2 (c)). The participants are required to perform 20 repetitions for each gesture under each scenario, where the gesture order is randomized for each scenario. This results in $3(scenarios) \times 13(gestures) \times 20(samples)$ =780 samples for each participant. The study was conducted in a quiet room with an average noise level of 31 dB.

During the study, participants were supplied with a Google Nexus 6P phone with our data collection app installed. For each gesture, they were instructed to click the start button on the app (Fig. 2 (b)). A gesture illustration appeared on the screen for reference, followed by a 3-second count-down for preparation. Then a

2-second blue dot animation was triggered as gesture guidance and the participants were instructed to follow the guidance to complete the gesture. The study lasted approximately 90 minutes for each participant. As a result, we collected 7111 valid gesture samples.

5.3 Gesture Classification

5.3.1 Generic Model. In this experiment, we first compared a set of machine-learning classifiers for classifying the gestures using different feature combinations. The purpose of this experiment is to determine an optimal feature combination strategy and a classifier that achieves the best performance across all scenarios. Specifically, we adopted 8-2 train-test data splitting strategy and experimented across 4 machine-learning models including Support Vector Machines (SVM), Random Forest (RF), Multilayer Perceptron (MLP), and Convolutional Neural Network (CNN). The results show that CNN outperformed other classifiers, especially using feature fusion (FusionCNN). In the sit scenario, FusionCNN achieved 94.55%; In the stand scenario, it reached 94.52%; In the walk scenario, FusionCNN had 86.14%, slightly behind SVM(RBF) (88.45%) using acoustic features. The results show that with our feature fusion technique, the recognition performance improves, especially showing more robustness under noisy situations such as in the walking scenario.

5.3.2 Leave-One-Out Experiments. To evaluate AirThumb's performance with new users, we conducted a leave-one-out test. In this test, data from one user was set aside as the test set, while data from the remaining 12 users was used for training. We repeated this process until all users were tested and averaged the performance for each classifier. The results show that CNN generally outperformed other classifiers (90.06% in sit scenario and 85.46% in walk scenario), except for the stand scenario (FusionCNN is 92.55%), where the MLP model with Fusion features achieved the highest accuracy (93.36%), significantly surpassing traditional machine learning methods. The confusion matrices of the best-performing models for each scenario in the leave-one-out tests are shown in Fig. 4.

5.3.3 Transfer Learning for the Personalized Model. In the real-world scenario, transfer learning has been widely used to adopt a general classification model to a personalized model using a small amount of user-specific data. To evaluate the adaptation of our approach, we further experimented adopting a transfer-learning strategy on the CNN model with a small amount of data from three left-out users. Specifically, we trained a model on 10 out of 13 users' data. We adopted similar a 8-2 train-test data splitting strategy for each gesture on the three left-out users. We then progressively increased the fine-tuned data sample of each gesture from their training set and test on their test set.

As shown in Fig. 5, adding user data progressively improved accuracy, showing an upward trend. In the sit and stand scenarios, adding user data steadily improved accuracy, with fusion features the trend will stabilize to a high value faster than others; in the walk scenario, accuracy showed overall improvement but fluctuated more, we attribute this to noise introduced by more dynamic motion, resulting in less stability. The result demonstrates that incorporating user-specific data enhances performance, and in future applications, user data can be added to a general model for quicker and better results.

5.3.4 Gesture Detection. We further evaluated the performance of the binary gesture detector and observed that the accuracy of the binary classification task was very high. In the general task, the accuracy of most feature-based CNN models was around 99%, except for imu-based CNN (IMUCNN) in the walk task, which only achieved 90.85%. During leave-one-out training, the accuracy reached a peak of 99.67%. CNN remained the most accurate model for both tasks. Overall, when switching from the sit and stand tasks to the walk task, accuracy decreased, with a more significant drop when using only IMU features. However, acoustic and fusion features still maintained high recognition accuracy.

6 LIMITATION AND FUTURE WORK

Although AirThumb achieves a promising mid-air thumb gesture recognition performance, there are several limitations that merit further improvements. Firstly, as a proof-of-concept, we conducted a series of offline experiments to evaluate the concept of AirThumb, while further real-time on-board implementation is necessary to enable AirThumb in real-world usages. Further online evaluation is also needed in our future work. Secondly, AirThumb is suffering from a relatively low accuracy in noise conditions such as in walking condition, which raises an implementation challenge for a robust real-time performance. In future works, we plan to explore more advanced de-noising techniques or recognition algorithms and evaluate with more data. Thirdly, as a proof-of-concept, we only implement and test AirThumb on one device. Since the hardware configurations (e.g. quality and location of the speaker and microphone)on different devices are various, we plan to deploy and evaluate our system on more devices.

7 CONCLUSION

In this work, we present AirThumb, a novel mid-air thumb-based interaction technique that leverages sensor fusion technique by combining acoustic sensing and IMU sensing. Through a two-stage gating gesture classification strategy, AirThumb detects 13 mid-air thumb gestures including 8 one-thumb gestures, 4 two-thumb gestures, and a default idle gesture across three scenarios. We evaluate AirThumb through two types of experiments. The generic model achieves recognition accuracies of 94.55%, 94.52%, and 86.14% under sitting, standing, and walking scenarios, respectively. The user-independent model shows the highest accuracies of 90.06%, 93.36%, and 85.46% under the same three scenarios. Although there are small drops of accuracies for the user-independent models, we show that AirThumb is capable to adapt on new users with fewer training data required using transfer learning.

ACKNOWLEDGMENTS

This research was partially supported by the Centre for Applied Computing and Interactive Media (ACIM) of School of Creative Media, City University of Hong Kong. This work was also partially supported by the National Natural Science Foundation of China (Project No. 62172346, No. 62402301), the CityU Contract Research Projects_RMGS (Project No. 9231581), the Innovation and Technology Fund (Project No. ITS/106/22), and the STU Scientific Research Initiation Grant (SRIG, Project No. NTF23024).

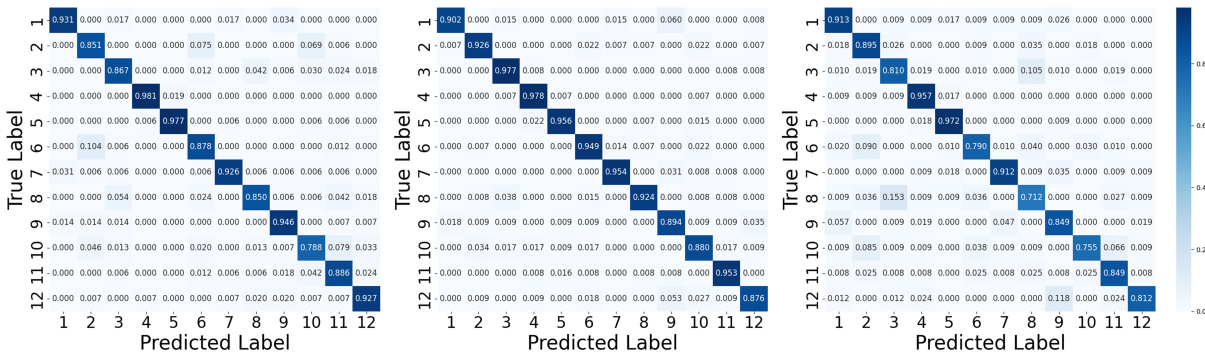

Figure 4: The leave-one-out evaluation averaged the confusion matrices, recording the best classifier's confusion matrix for each scenario: (a) sit task, FusionCNN; (b) stand task, FusionMLP; (c) walk task, FusionCNN.

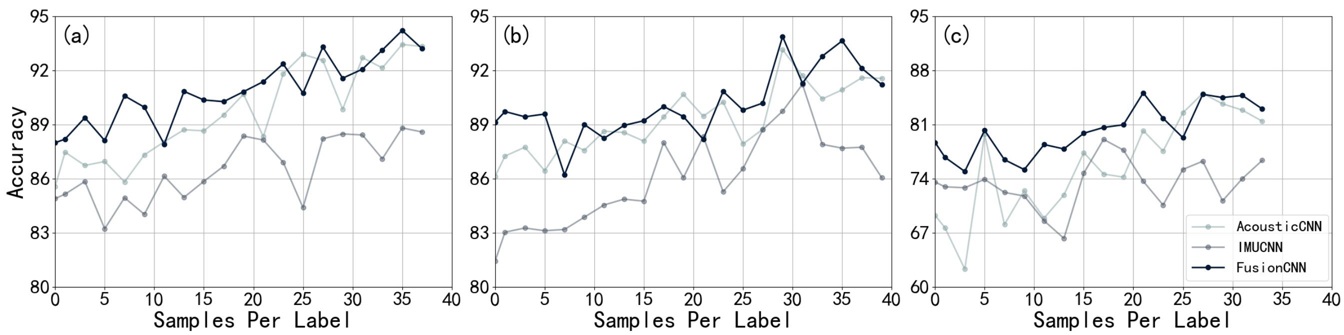

Figure 5: The results of transfer learning on the CNN model demonstrate the accuracy curves of different feature-based CNNs across various task scenarios: (a) sit scenario, (b) stand scenario, and (c) walk scenario.

Table 2: Generic model result on 12 gestures classification.

Scenario	Feature	SVM (Linear)	SVM (Poly)	SVM (RBF)	RF	MLP	CNN
Sit	Acoustic	84.07%	81.76%	85.12%	77.36%	87.21%	**93.92%**
Sit	IMU	83.65%	85.95%	83.44%	87.84%	81.76%	**88.47%**
Sit	Fusion	90.99%	90.57%	91.40%	92.24%	91.82%	**94.55%**
Stand	Acoustic	88.60%	88.60%	89.91%	87.28%	88.82%	**93.20%**
Stand	IMU	86.62%	88.60%	85.75%	**89.25%**	79.82%	87.72%
Stand	Fusion	93.20%	93.42%	91.67%	93.42%	91.23%	**94.52%**
Walk	Acoustic	87.76%	86.61%	**88.45%**	80.83%	76.91%	85.91%
Walk	IMU	72.29%	75.06%	77.37%	**79.45%**	68.82%	72.98%
Walk	Fusion	84.29%	84.98%	85.91%	85.68%	85.21%	**86.14%**

REFERENCES

[1] Shiri Azenkot and Shumin Zhai. 2012. Touch behavior with different postures on soft smartphone keyboards. In *Proceedings of the 14th International Conference on Human-Computer Interaction with Mobile Devices and Services* (San Francisco, California, USA) *(MobileHCI '12)*. Association for Computing Machinery, New York, NY, USA, 251–260. https://doi.org/10.1145/2371574.2371612

[2] Patrick Baudisch and Gerry Chu. 2009. Back-of-device interaction allows creating very small touch devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Boston, MA, USA) *(CHI '09)*. Association for Computing Machinery, New York, NY, USA, 1923–1932. https://doi.org/10.1145/1518701.1518995

[3] Joanna Bergstrom-Lehtovirta and Antti Oulasvirta. 2014. Modeling the functional area of the thumb on mobile touchscreen surfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 1991–2000. https://doi.org/10.1145/2556288.2557354

[4] Ke-Yu Chen, Daniel Ashbrook, Mayank Goel, Sung-Hyuck Lee, and Shwetak Patel. 2014. AirLink: sharing files between multiple devices using in-air gestures. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (Seattle, Washington) *(UbiComp '14)*. Association for Computing Machinery, New York, NY, USA, 565–569. https://doi.org/10.1145/2632048.2632090

[5] Taizhou Chen, Tianpei Li, Xingyu Yang, and Kening Zhu. 2023. EFRing: Enabling Thumb-to-Index-Finger Microgesture Interaction through Electric Field Sensing Using Single Smart Ring. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 4, Article 161 (Jan. 2023), 31 pages. https://doi.org/10.1145/3569478

[6] Taizhou Chen, Lantian Xu, Xianshan Xu, and Kening Zhu. 2021. GestOnHMD: Enabling Gesture-based Interaction on Low-cost VR Head-Mounted Display. *IEEE Transactions on Visualization and Computer Graphics* 27, 5 (2021), 2597–2607. https://doi.org/10.1109/TVCG.2021.3067689

[7] Céline Coutrix and Camélia Prost. 2024. Impact of Fingernails Length on Mobile Tactile Interaction. In *Proceedings of the 2024 CHI Conference on Human Factors*

in Computing Systems (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 553, 21 pages. https://doi.org/10.1145/3613904.3642037

[8] Rachel Eardley, Anne Roudaut, Steve Gill, and Stephen J. Thompson. 2017. Understanding Grip Shifts: How Form Factors Impact Hand Movements on Mobile Phones. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 4680–4691. https://doi.org/10.1145/3025453.3025835

[9] Mayank Goel, Jacob Wobbrock, and Shwetak Patel. 2012. GripSense: using built-in sensors to detect hand posture and pressure on commodity mobile phones. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology* (Cambridge, Massachusetts, USA) *(UIST '12)*. Association for Computing Machinery, New York, NY, USA, 545–554. https://doi.org/10.1145/2380116.2380184

[10] Jun Gong, Yang Zhang, Xia Zhou, and Xing-Dong Yang. 2017. Pyro: Thumb-Tip Gesture Recognition Using Pyroelectric Infrared Sensing. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (Québec City, QC, Canada) *(UIST '17)*. Association for Computing Machinery, New York, NY, USA, 553–563. https://doi.org/10.1145/3126594.3126615

[11] Khalad Hasan, David Ahlström, Junhyeok Kim, and Pourang Irani. 2017. AirPanes: Two-Handed Around-Device Interaction for Pane Switching on Smartphones. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 679–691. https://doi.org/10.1145/3025453.3026029

[12] Khalad Hasan, Junhyeok Kim, David Ahlström, and Pourang Irani. 2016. Thumbs-Up: 3D Spatial Thumb-Reachable Space for One-Handed Thumb Interaction on Smartphones. In *Proceedings of the 2016 Symposium on Spatial User Interaction* (Tokyo, Japan) *(SUI '16)*. Association for Computing Machinery, New York, NY, USA, 103–106. https://doi.org/10.1145/2983310.2985755

[13] Ken Hinckley, Seongkook Heo, Michel Pahud, Christian Holz, Hrvoje Benko, Abigail Sellen, Richard Banks, Kenton O'Hara, Gavin Smyth, and William Buxton. 2016. Pre-Touch Sensing for Mobile Interaction. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 2869–2881. https://doi.org/10.1145/2858036.2858095

[14] Xinhui Jiang, Yang Li, Jussi P.P. Jokinen, Viet Ba Hirvola, Antti Oulasvirta, and Xiangshi Ren. 2020. How We Type: Eye and Finger Movement Strategies in Mobile Typing. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3313831.3376711

[15] Amy K. Karlson and Benjamin B. Bederson. 2007. ThumbSpace: Generalized One-Handed Input for Touchscreen-Based Mobile Devices. In *Human-Computer Interaction – INTERACT 2007*. Springer Berlin Heidelberg, Berlin, Heidelberg, 324–338.

[16] Amy K. Karlson, Benjamin B. Bederson, and José Luis Contreras-Vidal. 2006. Understanding single-Handed mobile device interaction. *University of Maryland* (01 2006). https://api.semanticscholar.org/CorpusID:18670765

[17] Kunihiro Kato and Kaori Ikematsu. 2023. Detecting Thumb-Posture for One-handed Interactions with Smartphone using Acoustic Sensing. In *Proceedings of the 2023 ACM International Symposium on Wearable Computers* (Cancun, Quintana Roo, Mexico) *(ISWC '23)*. Association for Computing Machinery, New York, NY, USA, 129–131. https://doi.org/10.1145/3594738.3611378

[18] Namhyun Kim and Jinkyu Lee. 2017. Towards grip sensing for commodity smartphones through acoustic signature. In *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers* (Maui, Hawaii) *(UbiComp '17)*. Association for Computing Machinery, New York, NY, USA, 105–108. https://doi.org/10.1145/3123024.3123090

[19] Namhyun Kim, Junseong Lee, Joyce Jiyoung Whang, and Jinkyu Lee. 2020. SmartGrip: grip sensing system for commodity mobile devices through sound signals. *Personal and Ubiquitous Computing* 24, 5 (2020), 643–654.

[20] Jianwei Lai and Dongsong Zhang. 2015. ExtendedThumb: A Target Acquisition Approach for One-Handed Interaction With Touch-Screen Mobile Phones. *IEEE Transactions on Human-Machine Systems* 45, 3 (2015), 362–370. https://doi.org/10.1109/THMS.2014.2377205

[21] Huy Viet Le, Sven Mayer, Maximilian Weiß, Jonas Vogelsang, Henrike Weingärtner, and Niels Henze. 2020. Shortcut Gestures for Mobile Text Editing on Fully Touch Sensitive Smartphones. *ACM Trans. Comput.-Hum. Interact.* 27, 5, Article 33 (Aug. 2020), 38 pages. https://doi.org/10.1145/3396233

[22] Xingyu Li, Hongjun Dai, Lizhen Cui, and Ya Wang. 2017. SonicOperator: Ultrasonic gesture recognition with deep neural network on mobiles. In *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*. IEEE, San Francisco, CA, USA, 1–7. https://doi.org/10.1109/UIC-ATC.2017.8397483

[23] Hyunchul Lim, Jungmin Chung, Changhoon Oh, SoHyun Park, Joonhwan Lee, and Bongwon Suh. 2018. Touch+Finger: Extending Touch-based User Interface Capabilities with "Idle" Finger Gestures in the Air. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. Association for Computing Machinery, New York, NY, USA, 335–346. https://doi.org/10.1145/3242587.3242651

[24] Kang Ling, Haipeng Dai, Yuntang Liu, Alex X. Liu, Wei Wang, and Qing Gu. 2022. UltraGesture: Fine-Grained Gesture Sensing and Recognition. *IEEE Transactions on Mobile Computing* 21, 7 (2022), 2620–2636. https://doi.org/10.1109/TMC.2020.3037241

[25] Fabrice Matulic, Taiga Kashima, Deniz Beker, Daichi Suzuo, Hiroshi Fujiwara, and Daniel Vogel. 2023. Above-Screen Fingertip Tracking with a Phone in Virtual Reality. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI EA '23)*. Association for Computing Machinery, New York, NY, USA, Article 18, 7 pages. https://doi.org/10.1145/3544549.3585728

[26] Rajalakshmi Nandakumar, Vikram Iyer, Desney Tan, and Shyamnath Gollakota. 2016. FingerIO: Using Active Sonar for Fine-Grained Finger Tracking. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 1515–1525. https://doi.org/10.1145/2858036.2858580

[27] Alexander Ng, Stephen A. Brewster, and John H. Williamson. 2014. Investigating the effects of encumbrance on one- and two- handed interactions with mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 1981–1990. https://doi.org/10.1145/2556288.2557312

[28] Ian Oakley and Doyoung Lee. 2014. Interaction on the edge: offset sensing for small devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 169–178. https://doi.org/10.1145/2556288.2557138

[29] Kseniia Palin, Anna Maria Feit, Sunjun Kim, Per Ola Kristensson, and Antti Oulasvirta. 2019. How do People Type on Mobile Devices? Observations from a Study with 37,000 Volunteers. In *Proceedings of the 21st International Conference on Human-Computer Interaction with Mobile Devices and Services* (Taipei, Taiwan) *(MobileHCI '19)*. Association for Computing Machinery, New York, NY, USA, Article 9, 12 pages. https://doi.org/10.1145/3338286.3340120

[30] Pekka Parhi, Amy K. Karlson, and Benjamin B. Bederson. 2006. Target size study for one-handed thumb use on small touchscreen devices. In *Proceedings of the 8th Conference on Human-Computer Interaction with Mobile Devices and Services* (Helsinki, Finland) *(MobileHCI '06)*. Association for Computing Machinery, New York, NY, USA, 203–210. https://doi.org/10.1145/1152215.1152260

[31] Yong S Park and Sung H Han. 2010. One-handed thumb interaction of mobile devices from the input accuracy perspective. *International Journal of Industrial Ergonomics* 40, 6 (2010), 746–756.

[32] Keith B. Perry and Juan Pablo Hourcade. 2008. Evaluating one handed thumb tapping on mobile touchscreen devices. In *Proceedings of Graphics Interface 2008* (Windsor, Ontario, Canada) *(GI '08)*. Canadian Information Processing Society, CAN, 57–64.

[33] Yang Qifan, Tang Hao, Zhao Xuebing, Li Yin, and Zhang Sanfeng. 2014. Dolphin: Ultrasonic-Based Gesture Recognition on Smartphone Platform. In *2014 IEEE 17th International Conference on Computational Science and Engineering*. IEEE, Chengdu, China, 1461–1468. https://doi.org/10.1109/CSE.2014.273

[34] Wenjie Ruan, Quan Z. Sheng, Lei Yang, Tao Gu, Peipei Xu, and Longfei Shangguan. 2016. AudioGest: enabling fine-grained hand gesture detection by decoding echo signal. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (Heidelberg, Germany) *(UbiComp '16)*. Association for Computing Machinery, New York, NY, USA, 474–485. https://doi.org/10.1145/2971648.2971736

[35] Chun Yu, Xiaoying Wei, Shubh Vachher, Yue Qin, Chen Liang, Yueting Weng, Yizheng Gu, and Yuanchun Shi. 2019. HandSee: Enabling Full Hand Interaction on Smartphone with Front Camera-based Stereo Vision. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3290605.3300935

[36] Sangki Yun, Yi-Chao Chen, Huihuang Zheng, Lili Qiu, and Wenguang Mao. 2017. Strata: Fine-Grained Acoustic-based Device-Free Tracking. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services* (Niagara Falls, New York, USA) *(MobiSys '17)*. Association for Computing Machinery, New York, NY, USA, 15–28. https://doi.org/10.1145/3081333.3081356

[37] Man Zhou, Qian Wang, Jingxiao Yang, Qi Li, Feng Xiao, Zhibo Wang, and Xiaofeng Chen. 2018. PatternListener: Cracking Android Pattern Lock Using Acoustic Signals. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security* (Toronto, Canada) *(CCS '18)*. Association for Computing Machinery, New York, NY, USA, 1775–1787. https://doi.org/10.1145/3243734.3243777

[38] Kening Zhu, Xiaojuan Ma, Haoyuan Chen, and Miaoyin Liang. 2017. Tripartite effects: exploring users' mental model of mobile gestures under the influence of operation, handheld posture, and interaction space. *International Journal of Human–Computer Interaction* 33, 6 (2017), 443–459.