MHTrack: mmWave-Based Mobile Hand Tracking

Xiulong Liu, *Senior Member, IEEE*, Hankai Liu, Yantao Han, Xin Xie, *Member, IEEE*, Xinyu Tong, and Keqiu Li, *Fellow, IEEE*

***Abstract*—Non-intrusive hand tracking with mmWave radar technology is important in various Human-Computer Interaction (HCI) scenarios. However, existing mmWave-based solutions require users to be stationary and restrict a fixed hand motion area, which limits application flexibility and user experience. This paper proposes a novel mmWave-based Mobile Hand Tracking (MHTrack) system, which tracks user's hand gestures during walking. MHTrack focuses on tracking both *absolute* hand trajectory in the global coordinate system and *relative* hand trajectory to the body. Specifically, we propose a wake-up mechanism for hand motion capture, in which hand point cloud can be recognized even under body interference and noise. We propose a hand tracking strategy named local spatial update, which overcomes the sparsity and instability of point clouds, to obtain absolute hand trajectory. Subsequently, we propose a hand anchor correction method to suppress anchor offset and remove the impact of body movement from absolute hand trajectory, thereby obtaining relative hand trajectory. As a case study, we project the relative hand trajectory onto a 2D image and feed it into a gesture recognition model to recognize the gestures. We conduct extensive experiments to evaluate the performance of MHTrack. Results demonstrate a 3D hand trajectory tracking error of 3.6 cm in an area of 3.2 m × 4.8 m and a gesture recognition accuracy of 99% with 30 gesture classes.**

***Index Terms*—mmWave radar, hand tracking, gesture recognition, wireless sensing.**

I. Introduction

A. Motivation & Problem Statement

MOBILE hand tracking and recognition have long been a popular topic in human-computer interaction. First, hand tracking and recognition can be applied in various scenarios, such as turning off a phone alarm or answering calls without picking up the phone [1], recognizing sign language used by people with hearing and speech differences [2], or swiping screens on a wall display [3]. There have been many solutions using wearing devices, computer vision, or wireless sensing techniques. Placing wearable devices such as Inertial Measurement Unit (IMU) [4], [5] and Surface Electromyography (sEMG) [6] on bodies may be uncomfortable for users. Computer vision-based methods [7] operate by obtaining clear visual images, which suffer from the risk of privacy leakage and do not function well in dark environments. In RF technology field, RFID has gained attention due to its low cost and simple structure [8], [9], but RFID tags are susceptible to interference, lack accuracy, and sometimes need to be attached to the sensing target. Besides, given the existing well-established infrastructure, many studies utilize WiFi's Channel State Information (CSI) for gesture sensing [10], [11], [12], [13]. However, due to the longer wavelength and lower spatial resolution of WiFi, environmental changes and the presence of other mobile individuals pose significant challenges to the sensing precision. Similar to WiFi, Bluetooth also faces the problem of low resolution caused by its low frequency. In contrast, mmWave radar is flexible in deployment and can offer better spatial resolution with its high frequency and large bandwidth, enabling more accurate gesture reconstruction. However, existing mmWave hand sensing techniques require users to remain physically stationary in one place and cannot cope with the moving body state, resulting in application limitations in user mobility scenarios. Overcoming this limitation can realize a more natural and comfortable interaction experience, which serves as our motivation. For example, in a typical living room scenario, one could perform a gesture mid-walk towards the balcony to open the curtains in advance, avoiding the cumbersome process of "walk-stop-perform gesture-continue walking". In summary, this paper studies the problem of mobile hand tracking using mmWave radar, which can be formally defined as follows:

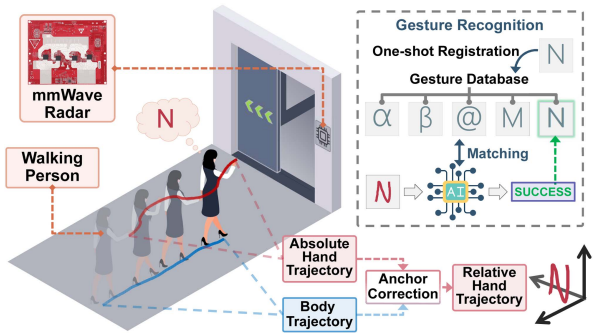


Fig. 1. Mobile hand tracking with MHTrack.

As illustrated in Fig. 1, a user moves one hand in the air while walking in a room setting. We collect mmWave data, and want to know (1) the absolute hand trajectory in the global coordinate system and (2) the relative hand trajectory to the body.

Received 17 September 2024; revised 27 February 2025; accepted 7 April 2025. Date of publication 11 April 2025; date of current version 6 August 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62372324 and Grant 62402336, and in part by the Innovation Capability Support Program of Shaanxi under Grant 2023-CX-TD-08. Recommended for acceptance by V. P. Ranga Rao. *(Corresponding author: Hankai Liu.)*

The authors are with the College of Intelligence and Computing, Tianjin University, Tianjin 300350, China (e-mail: xiulong_liu@tju.edu.cn; hkliu@tju.edu.cn; hyt@tju.edu.cn; xinxie@tju.edu.cn; xytong@tju.edu.cn; keqiu@tju.edu.cn).

Digital Object Identifier 10.1109/TMC.2025.3560074

1536-1233 © 2025 IEEE. All rights reserved, including rights for text and data mining, and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.

B. Limitations of Prior Art

The works closely related to this paper can be generally classified into two categories: hand tracking [14], [15], [16], [17] and gesture recognition [18], [19], [20], [21], [22]. Generally, these methods have the following limitations. (1) Hand tracking methods usually restrict hands to a fixed small area near mmWave radar to prevent the hand point cloud from being overwhelmed by interference such as body point cloud, which, however, renders inconvenience for users. (2) Gesture recognition methods usually rely on signal signatures such as point cloud, temporal spectrum and heatmap streams, which are very sensitive to the environment. Hence, they often require extensive data acquisition and additional training to adapt to new gestures or environments. (3) A common limitation of these two categories of methods is that none of them can support the cases of mobile users. Additionally, human imaging methods [23], [24] based on deep learning techniques can effectively estimate human body morphology and can also be used for hand tracking. However, these methods have the following limitations: (1) In terms of development workflow, they require a complex and time-consuming training process. (2) Regarding deployment generalization, their performance in new environments may decline due to insufficient generalization capabilities. (3) In terms of algorithm performance, they focus more on the overall accuracy of all body joints, lacking precise tracking of the hand joint.

C. MHTrack in a Nutshell

To overcome the limitations of existing work, we propose a mmWave-based Mobile Hand Tracking system, MHTrack, which supports user movement and one-shot learning of new gestures. The basic framework of MHTrack consists of the following two modules. (1) *Absolute hand tracking:* We first generate Range-Azimuth (RA) point clouds and use them to implement standard point target tracking to obtain the user's body movement trajectory. During body tracking, when the user moves one hand in the air, a hand point cloud appears around the body. At this moment, the hand tracking module is awakened to capture the hand movement. The resultant sequence of hand tracking represents the hand's absolute positions in space, which we call the *absolute hand trajectory*. (2) *Relative hand tracking:* To remove extra hand displacement caused by body movement, we remove the body trajectory from the absolute hand trajectory to track the *relative hand trajectory*, which represents the hand position change relative to the user and can be understood as a gesture. Then, we project the relative hand trajectory onto a panel to obtain a 2D gesture pattern and take gesture recognition as a case study, which achieves quick extension for new gestures by building a two-branch siamese network.

D. Challenges & Solutions

We need to address the following three major technical challenges when implementing the MHTrack system.

The first challenge is how to accurately capture hand data with unknown hand motion timing. The hand tracking module may be incorrectly woken up due to ghosting point clouds caused by noise. For this practical issue, many works have proposed solutions, such as finding motion frames by monitoring the presence of dynamic objects in the radar detection range [18], or setting energy thresholds based on the reflection strength of predefined gestures [25]. However, none of these existing methods can cope with the scenario where the user is moving and the gesture category is unknown. To this end, we propose a new method called credibility-based wake-up detection. Compared to noise, reflections from human hand should be more stable. Therefore, when a point cloud frequently appears around the human body and satisfies a predefined criterion, we consider it to be a hand. Then, through a state transition mechanism that includes four states and two conditions, we determine the starting and ending points of a gesture during hand tracking, which enables us to extract meaningful gesture data from the continuous stream of mmWave data.

The second challenge is how to accurately update the hand position with dynamic and interference-prone radar detection. Due to the instability and sparsity of point clouds generated by mmWave radar, close proximity of the hand to the body can result in the mixing of their point clouds. Regardless of whether the body is in motion, the point cloud continuously changes, making it nearly impossible to regard the human body as a rigid object and recognize hand motion by a background subtraction method. The dynamic reflections of human body severely disrupt the continuity of the hand point cloud. To solve this issue, we propose a local tracking strategy based on spatial update. We define a local candidate space near the human body with reference to the historical hand position, and extract the highest-intensity point in the space for stable and continuous hand tracking.

The third challenge is how to correct gesture offset caused by non-ideal movement. We define the hand anchor as the center around which the person performs a gesture. The offsets contain two types: anchor offset due to non-ideal human movement, and projection offset due to direction deviation in performing the gesture. For the former, it is a new problem arising in scenarios involving human movement. Due to the offset between the macroscopic motion trajectories of the body and hand, directly subtracting the body trajectory from the absolute hand trajectory cannot accurately yield the relative hand trajectory. We propose an anchor correction method. Based on radar position and body trajectory, we infer a series of hand anchors to calculate relative hand coordinates, and then correct all anchors to get the gesture, $i.e.$, relative hand trajectory. For the latter, we search for the optimal 2D virtual panel by maximizing the relative trajectory's projection area.

E. Novelty & Advantages

To the best of our knowledge, this paper is the first to study mmWave-based single-hand tracking for mobile users. The novelty of our work is mainly in the following three aspects: (1) Through the credibility-based wake-up detection and state transition mechanism, we overcome the noise interference and the uncertainty of gesture execution timing, realizing effective hand motion monitoring and extraction. (2) We propose the local spatial update and anchor correction techniques to mitigate

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.

TABLE I
COMPARISON OF HAND TRACKING METHODS

Method	Technigue	Range	Error	Dimension	Mobility
EasyTrack [10]	WiFi	1.5m	0.9cm	2D	NO
CentiTrack-3D [11]	WiFi	1.5m	2.9cm	3D	NO
mmWrite [16]	mmWave	3m	4cm	2D	NO
AirWrite [17]	mmWave	3m	1.6cm	2D	NO
MHTrack	**mmWave**	**4m**	**3.6cm**	**3D**	**YES**

TABLE II
COMPARISON OF GESTURE RECOGNITION METHODS

Method	Range	Category	Accuracy	One-shot Learning	Mobility
DI-Gesture [18]	1m	7	98.56%	NO	NO
Xie *et al.* [19]	2.4m	19	82.18%	NO	NO
Pantomime [3]	3m	21	94.05%	NO	NO
Wu *et al.* [20]	3m	10	96.5%	NO	NO
Tesla-Rapture [21]	4m	21	95.95%	NO	NO
mTransSee [22]	4.8m	5	96.7%	NO	NO
MHTrack	**4m**	**30**	**99%**	**YES**	**YES**

point cloud instability and gesture offset caused by non-ideal movement, achieving reliable hand tracking while the user is moving. (3) We construct a two-branch siamese network to recognize gestures, enabling model training with zero radar data and one-shot learning for new gestures. Our MHTrack system has the following main advantages over previous work: (1) Compared to existing hand tracking methods, MHTrack does not restrict a fixed area of hand motion and can well support user mobility with a hand tracking error of 3.6 cm within 4 m distance to radar, enhancing the deployment flexibility and user experience. A more thorough comparison of state-of-the-art hand tracking methods is shown in Table I.

(2) Compared to existing gesture recognition methods, MHTrack can quickly adapt to a new gesture with only one data sample, which can greatly reduce the data collection burden on users for registering new gestures. In the experiment of registering 30 new gestures, the recognition accuracy is 99%, which also proves the superior tracking performance of MHTrack. A more thorough comparison of state-of-the-art gesture recognition methods is shown in Table II.

(3) Compared to deep learning-based human imaging methods, MHTrack's hand tracking does not require model training, demonstrating excellent generalization capability and enabling plug-and-play functionality in various scenarios. Moreover, MHTrack fully utilizes the high spatial resolution of mmWave radar to maintain the precision and continuity of hand trajectory, achieving superior hand-tracking performance. In summary, MHTrack can realize outstanding performance in multiple dimensions simultaneously, including sensing range, tracking error, recognition accuracy, new gesture extension and user mobility support, which has not been achieved in existing works.

II. SYSTEM OVERVIEW

This paper mainly addresses a practical need to implement mmWave hand tracking while the user is moving. To this end, the MHTrack system is proposed and it consists of two parts: a mmWave radar and a backend server. The radar continuously monitors the sensing area and returns the collected data to the backend server. Then, the backend server uses the collected data to track the absolute hand trajectory in the air or further convert it into a relative hand trajectory, which is determined by different sensing tasks.

MHTrack for absolute hand tracking: As shown in Fig. 2, in the stage of data collection, the mmWave radar mixes the Received signals (Rx) with the Transmitted signals (Tx) to obtain the Intermediate Frequency (IF) signal, which is sampled and sent to the sensing terminal for processing. The sensing terminal first performs standard point target tracking based on RA point clouds to capture the user's body movement trajectory. Concurrently with body tracking, we design a credibility-based wake-up scheme to avoid noise interference and correctly activate the hand tracking. Then, we propose a local spatial update strategy to search for hand position, which overcomes the disappearance of the hand point cloud caused by the sparsity and instability, achieving continuous and stable hand tracking.

MHTrack for relative hand tracking: The purpose for obtaining the relative hand trajectory, i.e., gesture pattern, is to treat it as a command for device interaction, and we assume it is executed on a 2D virtual plane. Based on the absolute hand trajectory, we can further track the relative hand trajectory by body movement removal. During this process, we propose an anchor correction method and a complete state transition mechanism to solve the problems of anchor offset and unknown timing of gesture execution. Finally, we propose an area-based projection method to obtain the 2D gesture pattern and take gesture recognition as a case study in which we realize user-configurable recognition by building a two-branch siamese network model.

Coordinate system definition: In the process of tracking hand movement, we define the following coordinate system in this paper: the radar horizontal plane is parallel to the ground in the description and experiment of this paper. Taking radar as the origin, the x-, y-, and z-axes are perpendicular to the radar vertical plane, antenna array plane, and ground, respectively. The definition of azimuth follows the right-hand screw rule along the z-axis.

III. ABSOLUTE HAND TRACKING

Tracking the absolute hand moving trajectory in the air has many application scenarios. For example, in a virtual reality game, the user's immersive experience can be enhanced by tracking the real hand trajectory to synchronously control the moving trajectory of virtual objects. In this section, we provide the detailed process of absolute hand trajectory tracking.

A. Body Tracking

To support user moving, we perform body tracking to respond to their position change, thereby achieving precise and reliable hand tracking.

1) Data Collection: The mmWave radar sensing relies on the Frequency Modulated Continuous Wave (FMCW) chirp signal and its frequency increases linearly with time. The Tx signal

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.

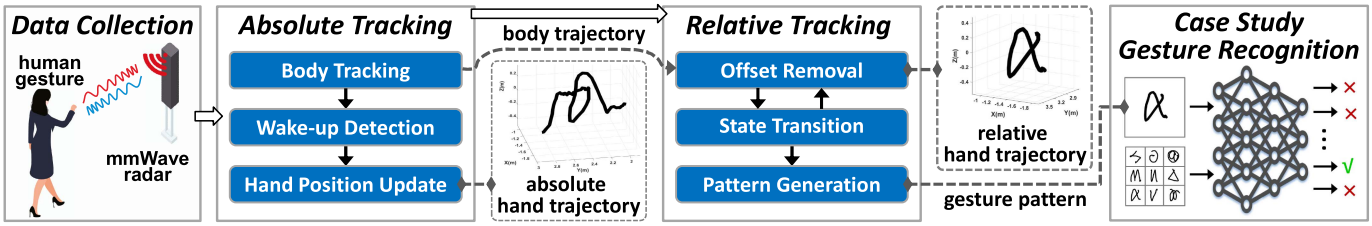


Fig. 2. Overview of the MHTrack system.

can be expressed as $S_t(t) = A\exp(j(2\pi f_0 t + \pi\frac{B}{T}t^2 + \varphi_0))$, where A, f_0, B, T and φ_0 respectively represent the amplitude, start frequency, sweep bandwidth, chirp duration, and phase offset of the Tx chirp signal. Assuming the distance between the target and the radar is R and the speed of light is c, the delay of the Rx signal relative to Tx signal is $\tau = 2R/c$. The Rx signal can be expressed as $S_r(t) = \alpha S_t(t-\tau)$, where α represents the signal attenuation factor. By mixing $S_r(t)$ and $S_t(t)$, we get the IF signal $S(t) = A_{IF}\exp(j(2\pi f_{IF}t + \varphi_{IF})) \approx A_{IF}\exp(j(2\pi\frac{2BR}{cT}t + \frac{4\pi R f_0}{c}))$, where A_{IF}, f_{IF} and φ_{IF} respectively represent the amplitude, frequency, and phase of the IF signal.

2) Human Detection. Range detection: Reflected signals from all distances within the sensing area are mixed, and targets at different distances cause different frequency components in the IF signal. We first apply a Fast Fourier Transform (FFT) to convert the signal from the time domain to the frequency domain, which is a process known as Range FFT. Then, we analyze the spectrum to determine the reflected intensity at different distances to obtain the information of target distance. During the process, we suppress spectral leakage caused by non-integer-period sampling by applying a Hanning window.

Static clutter removal: Reflections from static objects can cause serious interference in detecting dynamic targets. Therefore, we employ the vector mean-canceling algorithm to filter out static clutters. The method first calculates the average value of each chirp in a frame, and then subtracts it from each chirp. While suppressing static targets, the method can preserve a high Signal-to-Noise Ratio (SNR) for dynamic targets, which is suitable for human detection.

Azimuth estimation: To obtain the specific position of the user, we also need azimuth information. In this paper, we use the Capon beamforming algorithm, which can accurately estimate the direction of signal sources and has a strong ability to suppress clutter interference. Given a distance R and a direction θ, we can calculate the signal intensity from θ by $P_{R,\theta} = 1/(\mathbf{a}_\theta^H \mathbf{C}_R^{-1} \mathbf{a}_\theta)$, where $\mathbf{C}_R$ is the covariance matrix containing signal and noise variance information corresponding to distance R, and $\mathbf{a}_\theta$ is the steering vector representing the phase difference of the signal from θ arriving at each antenna unit. Specifically, for a uniform linear array with half-wavelength spacing and N channels, $\mathbf{a}_\theta = [1, \exp(-j\pi\sin\theta), \ldots, \exp(-j\pi(N-1)\sin\theta)]^T$. By scanning the azimuths of interest and calculating the intensity for each RA bin, we obtain the Range-Azimuth Map (RAM).

Point cloud generation: Next, the Cell Averaging Constant False Alarm Rate (CA-CFAR) algorithm is applied to generate RA point clouds from RAM. The algorithm creates a 2D window around each RA bin and calculates a threshold based on the signal intensity of other bins within the window to adaptively detect whether the target bin has a reflected object. CA-CFAR does not require preset thresholds, which makes it adaptable to different background noise conditions and thus improves the robustness of target detection.

Point cloud clustering: Considering that the distribution of human point clouds at different distances and angles is somewhat different, we choose the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm for point cloud clustering. DBSCAN does not need to know in advance the number of classifications, *i.e.*, the number of clusters, but rather groups points based on their density relationships in space, which is ideal for human body point clouds of different shapes and sizes. In this paper, we set the scanning radius to 0.25 m and the minimum number of included points to 14 for body clustering, and then we calculate the centroid of each cluster with point cloud intensity used as the weight.

3) Centroid Association: Performing human detection at each frame, we can obtain the human position at each moment. When there are multiple persons in the sensing environment, it's necessary to associate point cloud clusters with trajectories, which boils down to an optimal assignment problem. In this paper, we first compute the distances between cluster centroids and last trajectories, and then use these distances as the cost matrix of Hungarian algorithm to find the minimum matching. Performing such association and position update in each frame, we obtain the movement trajectories of multiple persons.

B. Tracking Wake-Up

As a person extends their hand, the point cloud of the hand appears around the body, and then the system starts tracking the hand. However, in addition to the hand, the scattered point cloud around the body may also be noise. To avoid wrongly waking up the system, we propose a credibility-based detection algorithm to distinguish between the hand and noise, which is implemented by the following three steps.

1) Re-Clustering: We assume that the hand is some distance away from the body during the gesture execution. At this time, hand reflection points are likely to appear as outlier points around the body in body clustering. We perform fine-grained re-clustering of these outlier points through DBSCAN with its scanning radius of 0.15 m and the minimum number of included points of 1. Then, we compute the centroids of these clusters obtained by re-clustering and designate them as candidate points.

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.

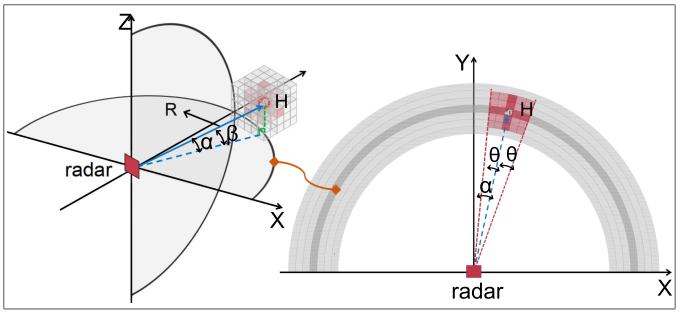


Fig. 3. Local spatial update.

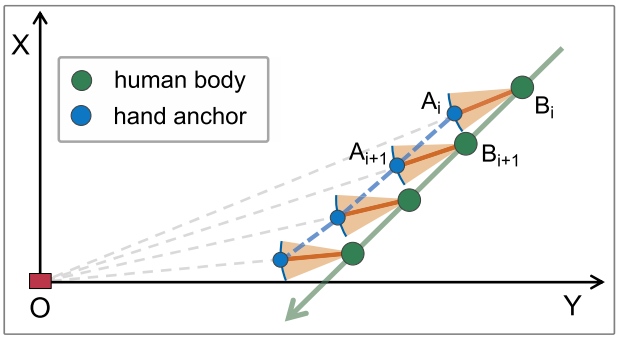


Fig. 4. Anchor offset.

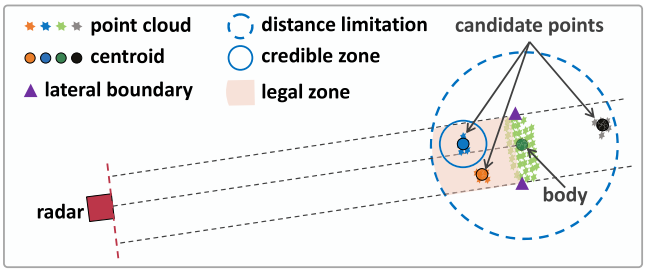


Fig. 5. The wake-up detection for hand tracking.

2) Legal Zone Establishment: To identify the hand among the candidate points obtained after re-clustering, we first filter them based on their positions. As shown in Fig. 5, we set a distance limitation of 0.8 m with the human body as the center, and then draw two boundary lines based on the body's lateral coverage, both of which are parallel to the radar-centroid line. Combining the lateral and distance limitations, we establish a legal zone. Then, we discard the candidate points outside it.

3) Credibility Check: For each candidate point within the legal zone, we define a credible zone as a circular area centered on the point with a radius of 0.2 m. In the next frame, we apply the same process to obtain new legal candidate points. If a new candidate point falls within the credible zone of a old candidate point, it replaces the old point and inherits its appearance count after being incremented by one. In our implementation, a point is considered as a hand if its appearance count reaches 3 within a time window of 5 consecutive frames.

Please note that, the legal zone is updated frame by frame as the body moves, and we update the credible zone with the latest candidate point. Therefore, users do not need to keep their bodies still to complete the wake-up process.

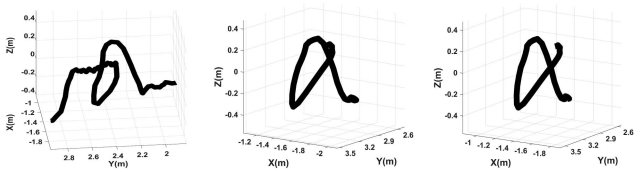
C. Hand Position Update

1) Local Spatial Update: Due to the interference of human movement and the surrounding environment, as well as the instability and sparsity of the point cloud itself, the hand point cloud may disappear during the hand tracking process, thus resulting in tracking errors. Therefore, we propose a tracking method named local spatial update. The method first generates a local space near the body based on the historical hand position, and then extracts the highest-intensity point within the space to achieve continuous update of hand position. As shown in Fig. 3, centered on the polar coordinate (range, azimuth, elevation) of the current hand position, we extend a local cube space in which we search for the hand position for the next frame. The establishment of the local space involves three dimensions (range, azimuth, elevation), and we extend two range bins in each dimension. Assuming the current hand position in spherical coordinates is (R, α, β), the new local space in three dimensions can be expressed as $[R-d, R+d]$, $[\alpha-\theta, \alpha+\theta]$, and $[\beta-\theta, \beta+\theta]$, where d represents the range extension for each side, and $\theta = \arctan(d/R)$ denotes the angular extension for each side. Based on the established local space, directly searching the hand position in it is a straightforward way, but traversing each spatial bin of the local space to search for the highest-intensity point requires performing more 2D beamforming, which is computationally burdensome and inefficient. Assuming the local space has dimensions of $n \times n \times n$, the time complexity of direct search is $O(n^3)$, while the two-step search reduces the complexity by an order of magnitude to $O(n^2+n)$. Therefore, we perform range-azimuth estimation followed by elevation estimation. In contrast, this approach can significantly reduce the computation while guaranteeing the localization accuracy.

2) Range-Azimuth Estimation: According to the range and azimuth settings of local space, we extract the results of Range FFT and re-perform 1D Capon to obtain local RAM. Then, we use the RA bin with the highest intensity in the local RAM as a new 2D position for the hand. There are two reasons for re-performing the Capon beamforming. First, different from the body which is a reflector with a large area, the reflection area of the hand is small and its reflection intensity is weak. Especially, when the hand moves slowly or is still, its reflection is easily filtered by static clutter removal. Therefore, we need to utilize the signal that is not processed by static clutter removal to calculate the RAM focusing on the hand. Second, hand tracking requires higher spatial resolution than body tracking, therefore we need to improve the angular granularity in beamforming to obtain a more accurate hand position.

3) Elevation Estimation: Then, we perform 2D Capon beamforming for the highest-intensity RA bin to estimate the hand's elevation. Compared to 1D Capon, the key difference lies in calculating the phase offset of the steering vector. Given a certain azimuth θ and elevation γ, when an array unit is located N_x and N_z half-wavelengths away from the reference unit in

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.



(a) Before body removal(b) w/o anchor correction (c) w/ anchor correction

Fig. 6. Hand trajectories obtained at different stages.

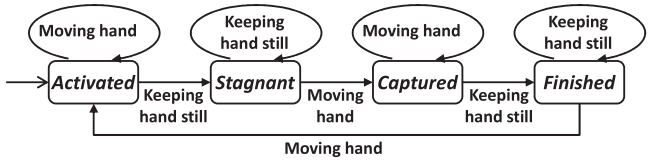


Fig. 7. State transition mechanism.

horizontal and vertical dimensions, respectively, its element in the steering vector is $\exp(-j\pi(N_x \sin\theta\cos\gamma + N_z \cos\theta \sin\gamma))$ We calculate the signal intensity for each elevation in the RA arc, and then select the highest-intensity elevation as the hand. Now, we can get the 3D coordinates of the hand by the transformation from the polar coordinate system to the Cartesian coordinate system.

IV. Relative Hand Tracking

Tracking the hand moving trajectory relative to users also has many application scenarios, such as controlling home devices through gesture recognition, and this section aims to obtain the relative hand trajectory.

A. Body Removal

To obtain the relative hand trajectory, we subtract the body displacement from the absolute hand trajectory. Assume that MHTrack starts to perform hand tracking from the i-th frame. We denote the body coordinate as B_i and the hand coordinate as H_i, and then the relative hand position of the i-th frame can be expressed as $G_i = H_i - B_i$. By performing such a subtraction operation on all points of the absolute hand trajectory, we can obtain the relative hand trajectory without body movement. However, non-ideal body movement may cause tracking errors.

As shown in Fig. 4, when the person's coverage azimuth significantly changes as they move, the movement trajectory of the hand anchor is inconsistent with the body, which can cause an offset of the relative trajectory. In order to overcome the offset, we propose a hand anchor correction method. The hand anchor is defined as the reference center when the user performs a gesture, i.e., the hand position when the user naturally reaches toward the radar. Denote the radar position as O and the length of the extended arm as L, which is determined by the distance between B_i and H_i. Then, we select a point A_i on the segment OB_i, located at a distance of L from B_i, as the hand anchor for the i-th frame. Thus, the trajectory offset caused by azimuth is included in A_i and can be eliminated by $G_i = H_i - A_i$. Based on the hand anchors, we can get a relative hand trajectory $\mathbf{G}_{ges} = [G_i, G_{i+1}, \ldots]$ without offset.

Taking "α" as an example, Fig. 6(a) shows the raw absolute hand trajectory, while Fig. 6(b) and (c) illustrate the relative hand trajectory before and after the anchor correction, respectively. We can visually appreciate the effective contribution of the proposed anchor correction method in addressing gesture offset and obtaining accurate gesture trajectories.

B. State Transition

In practice, the time interval during which a person performs the target gesture is unknown. In order to identify the starting and ending points of a gesture, MHTrack requires the user's hand to pause 0.5 s in the air before and after performing the gesture. Note that the pause is measured through the relative hand trajectory and is not affected by the body movement, allowing the user to perform this action while walking. We define two hand conditions, **still** and **move**, which are judged by calculating the coordinate variance of the relative hand trajectory. We use a time window of 0.5 s to calculate the coordinate variances along the x, y, and z axes, respectively. When the three variances are all less than a threshold, the hand is considered to be still, otherwise it is moving. We design a state transition mechanism to monitor the hand movement state. The mechanism is depicted in Fig. 7, which includes the following four states.*Activated:* After being woken up, MHTrack enters this state to start tracking the hand. *Stagnant:* When the hand keeps relatively still, the system enters this state to indicate that the user is about to start performing a gesture. *Captured:* The system enters this state from stagnant state when there is a significant displacement of the relative hand position. *Finished:* When the hand keeps relatively still again, indicating that the gesture execution is finished.

C. Gesture Pattern Generation

1) Trajectory Projection: In order to obtain a gesture pattern from the relative hand trajectory, we need to project the trajectory onto a virtual panel. We calculate the geometric center C of the relative hand trajectory and thus obtain a vector $\mathbf{v} = O - C$ connecting the hand trajectory to the radar position O. Then, the relative hand trajectory is projected onto a plane which is perpendicular to $\mathbf{v}$, and we obtain a 2D gesture trajectory by coordinate transformation, which we call the gesture pattern.

2) Optimal Projection Search: The direction deviation when performing a gesture may cause projection offset, *e.g.*, the deviation of the normal vector of the actual drawing plane from v, leading to distortion of the gesture pattern. For this reason, we propose an area-based approach to search for the optimal projection, the basic idea of which is to maximize the area of the projected gesture pattern. We use $\mathbf{v}$ as a reference to traverse different normal vector directions within a small angle range to obtain candidate projection panels. For the 2D gesture pattern obtained in each projection direction, we calculate the product of its horizontal and vertical spans to represent its area, and then select the one with the maximum area as the best. For the gesture

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.

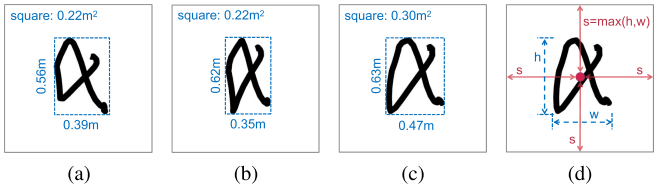


Fig. 8. Illustration of the image processing. In the optimal projection searching, (c) is better than (a) and (b). (d) shows the image standardization.

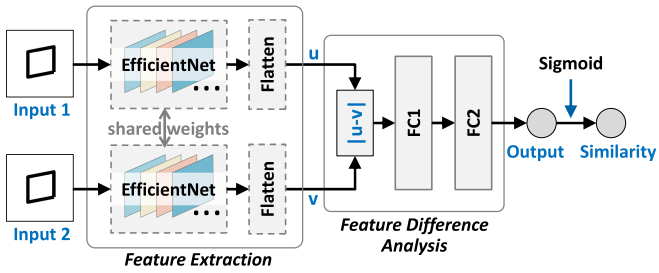


Fig. 9. Structure of ESNet for gesture recognition.

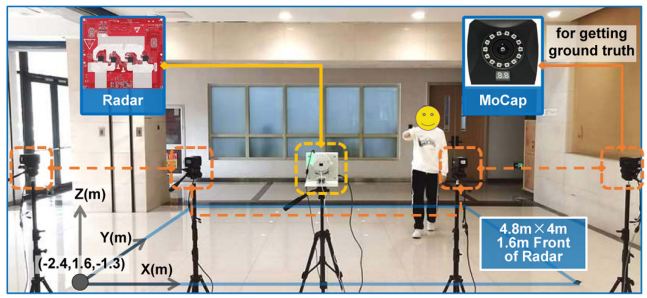


Fig. 10. Experimental environment.

"α", Fig. 8(a)∼(c) illustrate some results during the process of optimal projection search, among which Fig. 8(c) is better.

3) Image Standardization: Personal habits, as well as the distance and angle at which the gesture is performed, may affect the size of the gesture pattern. To reduce the difference between the same gestures, we achieve gesture standardization through image cropping. As shown in Fig. 8(d), we crop the image according to the size and geometric center of the gesture pattern to ensure the pattern always lies in the center region of the image regardless of its actual size. The standardization effect can also be visualized in Fig. 8(a)∼(c).

D. Gesture Recognition

Based on the gesture pattern, we can implement many applications. In this paper, we choose gesture recognition as a case study. Traditional supervised Convolutional Neural Network (cNN) pipline with a single sample input and fixed output categories can recognize well, but it needs to collect numerous training samples for each new gesture pattern. In order to support users to quickly register the gestures they want to recognize, we build a two-branch siamese model ESNet (EfficientNet & Siamese Network) with excellent gesture scalability. We use EfficientNet with shared weights as a feature extractor to get the features of two inputs, and then analyze the feature differences through multiple Fully Connected layers (FC) to determine whether they are the same. The ESNet serves user's personalized gesture needs through two stages: (1) One-shot Gesture Registration: for a new gesture, the user only needs to perform it once to complete the following registration process: MHTrack collects the mmWave signal data to generate a gesture image, and puts the image into the gesture database. (2) Gesture Recognition: for the new gesture performed again later, MHTrack generates a gesture image and searches the gesture database by calculating the similarity to get the recognition result. Next, we introduce the EfficientNet, siamese network and specific implementation details of ESNet.

EfficientNet [26]: is an efficient cNN architecture that can achieve good performance across various tasks and computational resource constraints. In our task, we modify its final fully connected layer to extract gesture features.

Siamese network [27]: is a network structure for metric learning. Its main idea is to learn the feature differences of sample pairs to determine their similarity. Such a learning way provides support for quick extension of new gestures.

Implementation details: As depicted in Fig. 9, ESNet receives two images as inputs which are respectively fed into EfficientNet for feature extraction, and two one-dimensional feature vectors u and v are obtained through the Flatten layer. Subsequently, the distance between corresponding elements of u and v is computed as the feature difference between the two inputs. Finally, this difference is processed through two FCs followed by a Sigmoid function to produce a numerical value between 0 and 1, which represents the similarity between the two inputs. We implement ESNet using the PyTorch framework. Regarding training parameters, we set the learning rate to 10^{-3}, optimizer to $Adam$, and loss function to $BCEWithLogitsLoss$. To alleviate the burden of collecting training data, we train our model using the Omniglot [28] dataset and partition it into a training set and a validation set in a ratio of 9:1. We ensure that the model has never seen mmWave gesture data during training. The training epoch is 20, and we choose the best epoch as the final model based on the performance of loss function and accuracy.

V. Performance Evaluation

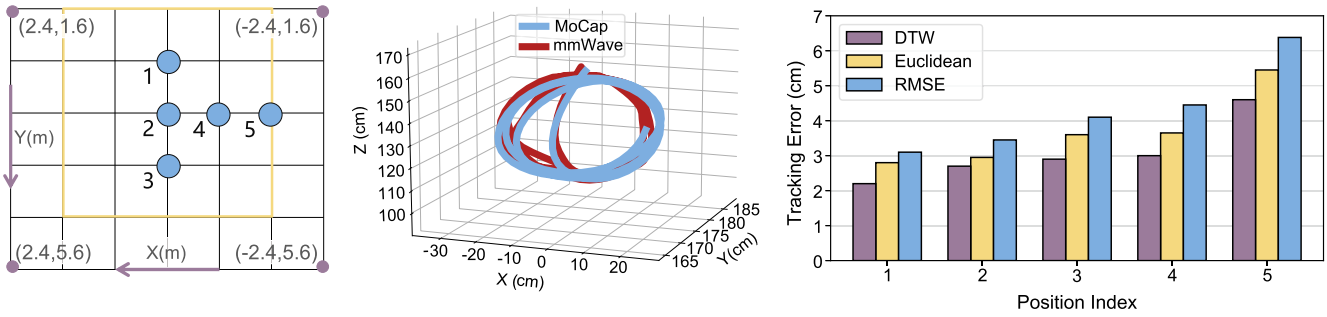
In this section, we conduct a series of experiments on both hand tracking precision and gesture recognition accuracy to evaluate the performance of the proposed MHTrack system. Each participant is fully informed about the experimental content and purpose.

A. Setup

1) Environment and Equipment. Environment: The default experimental environment is shown in Fig. 10.

We utilize floor tiles with a side length of 0.8 m as a reference for experimental design. The overall sensing area is $4.8\ \mathrm{m} \times 4\ \mathrm{m}$ and the distance between the radar and the sensing area is 1.6 m. The antenna array is positioned at a height of 1.3 m and

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.



(a) Five standing positions (b) Example of hand trajectory (c) Tracking errors

Fig. 11. The hand tracking performance of MHTrack when the subject stands still.

TABLE III
MMWAVE RADAR PARAMETERS

Parameter	Start Freq	Freq Slope	Ramp Time	Sample Rate
Value	$77GHz$	$100MHz/us$	$40us$	$7.2MHz$
Parameter	Sample Freq	Chirp Peroid	Chirp Num	Frame Period
Value	256	$50us$ (×6Tx)	128 (×6Tx)	$40ms$

vertical to the ground. Four Motion Capture (MoCap) cameras are positioned on either side of the radar at the same height as the radar and spaced 1.2 m apart.

Hardware: (1) mmWave radar: The TI MMWCAS-RF-EVM [29] is deployed for mmWave radar signal collection. The enabled antenna array is 6Tx8Rx, and the specific configuration parameters are listed in Table III.

(2) MoCap device: We deploy four Pluto 1.3C [30] MoCap cameras, which provide a 3D tracking precision of ≈ 1mm, to obtain the ground truth of hand tracking. The coordinate systems of radar and MoCap are aligned by the L-BFGS-B optimization algorithm.

Software: The data collection for the mmWave radar and MoCap is controlled by the official software, mmWave Studio and NOKOV, respectively. Subsequent data processing algorithms and network models are implemented through MATLAB R2022b and PyTorch, respectively.

2) Performance Metric. Tracking precision: We first focus on the tracking precision of 3D absolute hand trajectory. The subject draws hand trajectories in a clenched-fist way and the retro-reflective marker of MoCap system is placed on the proximal phalanx of the middle finger to obtain the ground truth. We calculate the tracking error by metrics of Dynamic Time Warping (DTW), Euclidean distance (Euclidean), and Root Mean Square Error (RMSE).

Recognition accuracy (Rec Acc): We also attach importance to the gesture pattern-based recognition. The recognition accuracy is defined as the ratio of the number of correctly recognized instances to the total number of instances. In addition, we show more detailed recognition results in the form of confusion matrices.

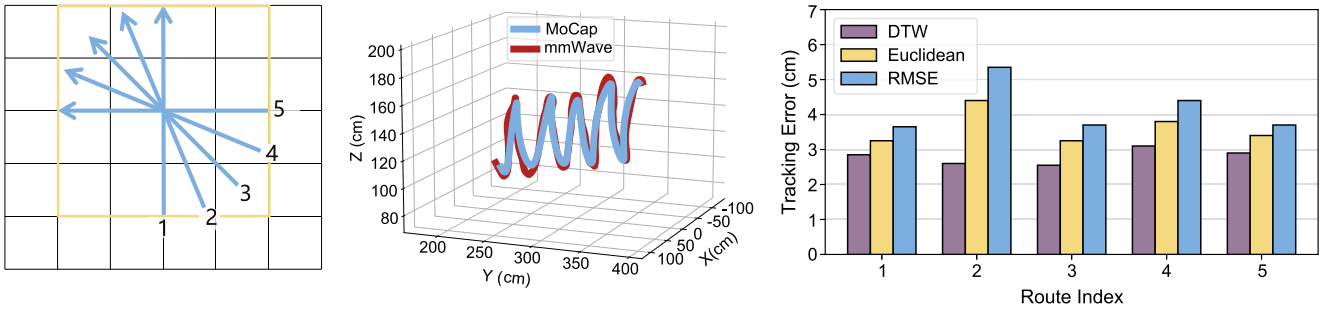
B. Evaluation of Tracking Precision

In this section, we first evaluate the tracking precision of absolute hand trajectory under different positions and human body states, and then validate the effectiveness of the proposed local tracking scheme. Due to the limited number and field of view of the MoCap camera, we restrict the activity region in 3.2 m × 3.2 m to ensure that the MoCap system offers stable and accurate ground truth. During one data collection, the subject draws circles in the air for 6 s. For each experimental setting, we collect data 10 times. We need the hand anchor to point towards the radar. For gesture interaction applications, this requirement is natural and easy to understand, posing no burden or inconvenience to users.

1) Performance in Different Positions: As shown in Fig. 11(a), we set up five positions to study the tracking precision. Fig. 11(b) visualizes one of the tracking results, and the median tracking error results under each setting are shown in Fig. 11(c). We can find that (1) with the increase of the distance and deviation angle between the radar and person, the distance error of the trajectory tends to increase but slightly. (2) The average errors of the three distance metrics, DTW, Euclidean and RMSE, are 3.1 cm, 3.7 cm, and 4.3 cm for all settings, respectively. Taking into account the resolution of the radar and the size of human hands, we believe the tracking precision is excellent and satisfactory.

2) Performance in Walking State: Next, we study the absolute hand tracking under walking state by setting up five routes, as shown in Fig. 12(a). Fig. 12(b) visualizes one of the tracking results. The tracking error results are shown in Fig. 12(c). We can find that (1) the average errors of the three distance metrics, DTW, Euclidean and RMSE, are 2.8 cm, 3.6 cm, and 4.2 cm for all settings, respectively. We compare these results with deep learning-based human imaging techniques. Compared to the approximately 5 cm Euclidean distance error recorded for hand joint by m3Track [24], MHTrack reduces the tracking error by 35%, demonstrating the performance advantage of our system. Through fully utilizing the high spatial resolution of mmWave radar, MHTrack maintains the accuracy and continuity of hand trajectory, achieving higher hand-tracking performance. (2) Whether walking directly toward the radar or in other directions, MHTrack maintains high precision. This indicates that the proposed methods, such as wake-up detection and spatial update,

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.



(a) Five walking routes (b) Example of hand trajectory (c) Tracking errors

Fig. 12. The hand tracking performance of MHTrack when the subject walks.

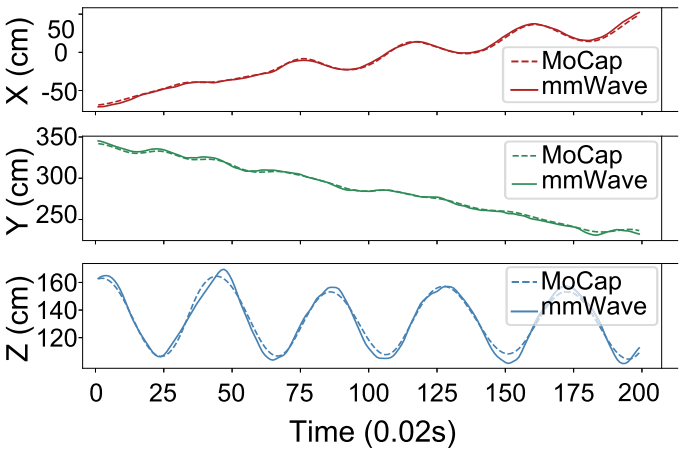


Fig. 13. Illustration of hand tracking result in three axes.

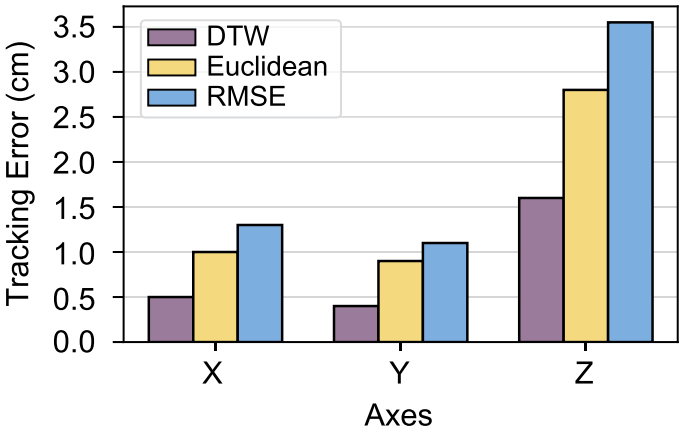


Fig. 14. Overall tracking performance.

can effectively separate the hand from the body, accurately identify the initial hand position, and achieve continuous and reliable hand tracking. We also evaluate the differences in the three coordinate axes, and the results are shown in Fig. 14.

The tracking error in the z-axis is larger relative to the x and y axes, which we attribute to the lower resolution in elevation of the radar equipment used for the experiment. Nonetheless, the tracking error in all axes is very small, which can also be intuitively seen in Fig. 13.

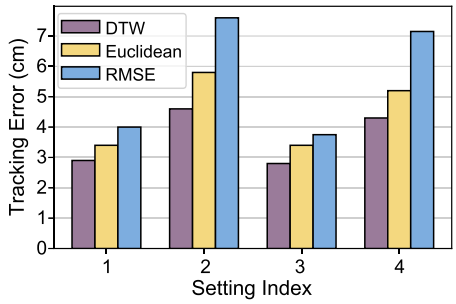


Fig. 15. Comparison between local and closest tracking.

3) Effectiveness Verification of Local Tracking: Existing hand tracking methods usually limit the body outside the sensing range in advance and localize the hand by extracting the entire point cloud within a fixed range, which is obviously not suitable for the case of human movement. The hand trajectory of MHTrack is obtained by local spatial update. In order to verify the effectiveness of the proposed method, we refer to the point selection scheme of mmWrite [16] and define a tracking method that continuously tracks the closest point to the radar in each frame of the point cloud within the sensing area, which we denote as the closest tracking. The tracking error of both methods is shown in Fig. 15.

Settings 1 and 3 correspond to the performance of local tracking under two body states, still and walking, respectively, while settings 2 and 4 correspond to the closest tracking. Taking the Euclidean distance as an example, tracking errors of the closest tracking grow by 70% and 53% relative to the local tracking under still and walking states respectively, which can prove the effectiveness and necessity of the local tracking strategy. Through adaptive spatial constraints, it can effectively filter out reflections from the environment and human body, enabling more accurate updates of hand positions.

4) Performance in Different Hand-Body Distance: During gesture execution, the proximity limit between the hand and body is a concern. To better clarify this limit, we set four different hand-body distances: 20 cm, 30 cm, 40 cm, and 50 cm. For each setting, we calculate the median Euclidean tracking error under both still and walking body states. The results are shown in Fig. 16. We can find that: (1) When the user is still

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.

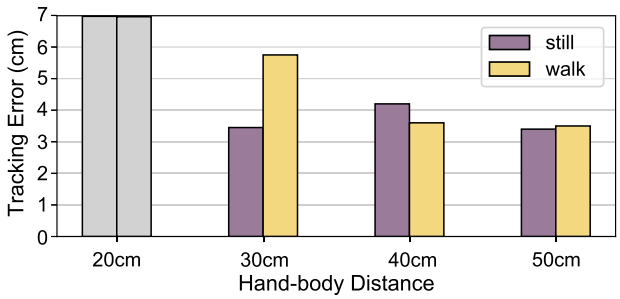
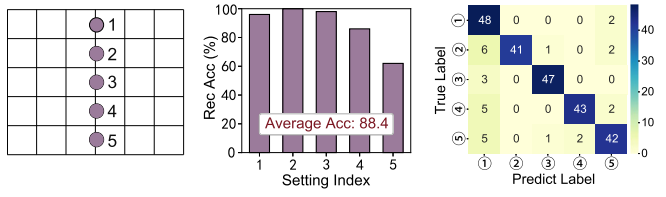


Fig. 16. Performance in different hand-body distance.

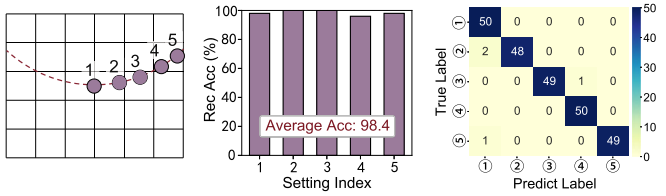


Fig. 17. Visualization of gesture database.



(a) Distance settings (b) Recognition results (c) CM of five gestures

Fig. 18. Impact of person-radar distance.



(a) Angle settings (b) Recognition results (c) CM of five gestures

Fig. 19. Impact of person-radar angle.

and the hand-body distance is no less than 30 cm, MHTrack can work well and the average error is only 3.7 cm, which is satisfactory. (2) When the user is walking and the hand-body distance is no less than 40 cm, MHTrack can achieve almost the same high precision as under a still state. Body movement significantly increases the dynamic information captured by the mmWave radar, causing more interference in extracting hand data, especially when the hand is close to the body. At a distance of 30 cm, the tracking error increases to 5.7 cm but can also meet the precise requirement for hand-tracking scenarios. (3) Due to the sparsity and instability of mmWave point clouds, a too-close distance between the hand and body causes serious point cloud mixing, resulting in tracking failure at 20 cm. However, this does not diminish the value of our system, as in regular gesture interaction scenarios, the hand-body distance is usually greater than 30 cm. In fact, in a natural and comfortable state, the hand-body distance when a person reaches their hand out is typically around 40 cm. At this distance, MHTrack can deliver excellent tracking performance no matter whether the user is stationary or moving.

C. Evaluation of Recognition Accuracy

Now we implement experiments within the sensing area of 4.8 m $\times$ 4 m to conduct an in-depth investigation of gesture recognition. Most experiments include five gestures, each of which is pre-collected only once at 2 m in front of the radar to construct the gesture database shown in Fig. 17.

For each experimental setting, we execute gestures 50 times (5 gestures $\times$ 10 times) for the recognition evaluation. In addition to the distance and angle, we also explore the impact of walking orientation, scenario, coexistence persons, and number of registered gestures.

1) Impact of Person-Radar Distance: In some cases, users need to perform gestures from a relatively long distance, while in other cases, users may want to be close to the device. However, the distance between the radar and person may have a significant influence on gesture capture and recognition. Therefore, learning about the recognition performance under different distance conditions is important. As shown in Fig. 18(a), we perform experiments at five positions directly in front of the radar, with the distances of 2 m, 2.8 m, 3.6 m, 4.4 m, and 5.2 m.

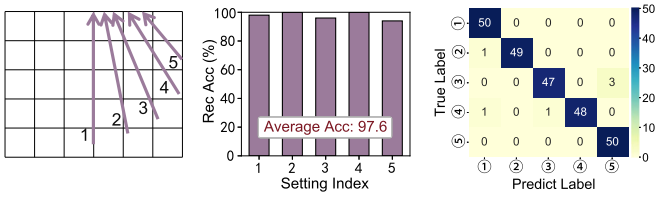
For every distance, each of the five gestures is executed 10 times, and a total of 250 gesture images are acquired. Fig. 18(b) reveals that the recognition accuracy at different distances is 88.4% on average, and Fig. 18(c) shows the confusion matrix (CM) of five gestures. From the figures, we can also find that: (1) When the subject performs gestures standing within 3.6 m, the recognition accuracy is very high, with an average of 98%. (2) The recognition accuracy gradually decreases when the subject stands further than 3.6 m. At 4.4 m, the recognition accuracy is 86%, and at 5.2 m, it is only 62%. We consider that there are two main reasons for the phenomenon: (1) the signal is continually attenuating during the propagation process, and the SNR is so low beyond a certain distance that it is hard to stably and efficiently extract the hand's movement trajectory from it. (2) The spatial resolution of the radar decreases as the distance increases, which means the details of the hand movement may not be accurately captured at a longer distance.

2) Impact of Person-Radar Angle: In order to ensure the user experience, achieving accurate gesture recognition at different angles is very important. As shown in Fig. 19(a), we perform experiments at 3.6 m distance with different angles, including 0°, 10°, 20°, 30° and 40°, which is sufficient for indoor gesture interactions.

Same as the data scales of distance experiments, every gesture is performed 10 times under each angle. Fig. 19(b) and (c) show the corresponding experimental results. We can find that MHTrack is robust to angles, and the average recognition accuracy under different angles is 98.4%.

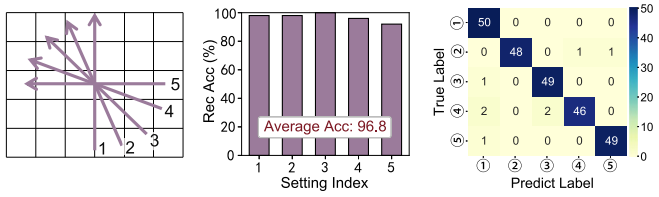
3) Impact of Walking Angle: The MHTrack appears to be effective when the subject is in a fixed position, including different distances and angles. However, in addition to being stationary in a specific location, users may also expect effective gesture control during walking. As depicted in Fig. 20(a), we let the five positions shown in Fig. 19(a) be dynamic.

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.



(a) Angle settings (b) Recognition results (c) CM of five gestures

Fig. 20. Impact of walking angle.



(a) Settings (b) Recognition results (c) CM of five gestures

Fig. 21. Impact of walking orientation.

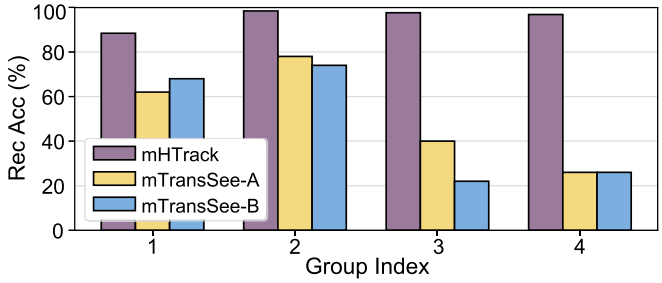
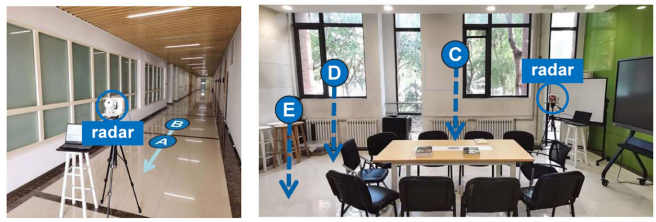


Fig. 22. Comparison between MHTrack and mTransSee.



(a) Corridor (b) Meeting room

Fig. 23. Experimental settings in more scenarios.

The subject walks towards the radar from different angles as a way to compare horizontally and vertically. The results depicted in Fig. 20(b) and (c) indicate that the MHTrack performs well and stably at five walking angles, with an average recognition accuracy of 97.6%, which is approximately the same as the results in Section V-C2.

4) Impact of Walking Orientation: This subsection further investigates gesture recognition under walking state. Unlike the last subsection, we do not ask the subject walking towards the radar. As shown in Fig. 21(a), the deviation angle of the five orientation settings gradually increases, among which setting 1 is directly in front of the radar antenna array, and the walking direction of setting 5 is perpendicular to the antenna array. The experimental results are shown in Fig. 21(b) and (c). The overall recognition accuracy for the five orientations is 96.8%, which is only a 0.8% decrease compared to the last subsection. Setting 5 has a slightly lower accuracy of 92%. We think the reason is that in the second half of setting 5, the radar is located at the back side of the subject's body, and it is a bit difficult to perform gestures at this time, which may cause a large deviation. In summary, MHTrack can provide a superior user experience compared to gesture interaction systems that only support a still state.

5) Performance Comparison With mTransSee: In the field of gesture recognition, mTransSee [22] is a representative state-of-the-art work with domain adaptation capability. For the gesture categories predefined during the model pretraining stage, it can adapt to variations of position or environment by fine-tuning the model with only 8 data samples. However, we are curious about the following situations: (1) For gestures that need to be recognized, does mTransSee remain effective if fewer data samples have been seen during the model training stage? (2) After fine-tuning the pre-trained model at a low cost, mTransSee can well adapt the predefined gestures in a new environment, but can it still adapt well to a new gesture with 8 samples? In response to these two questions, we re-implement mTransSee and train eight models on the four groups of experiments shown in Fig. 18(a) $\sim$ Fig. 21(a). Each group includes two mTransSee models, A and B. Without loss of generality, let's take group 1 as an example to specifically describe the implementation and testing process of the two models: (1) mTransSee-A: Based on all the newly collected gesture samples in group 1, we use 80% of them for model training and 20% for model testing. (2) mTransSee-B: A pre-trained model with good performance is first obtained through a large amount of radar data and model implementation details provided by the author, then the model is fine-tuned by 200 new gesture samples ($=$ 5 distances $\times$ 5 gestures $\times$ 8 samples), and finally testing the model through the remaining 50 gesture samples ($=$ 5 distances $\times$ 5 gestures $\times$ 2 samples). All results for the four setting groups are shown in Fig. 22, from which we can find that mTransSee's effectiveness depends on a considerable amount of radar data, making it hard to train a model with outstanding recognition performance based on only a few hundred samples. Additionally, mTransSee does not have the ability to quickly adapt to new gestures, especially in the walking case. We emphasize that for each gesture, only one sample is collected as a reference before testing MHTrack, and the sample has never been seen during model training stage, demonstrating the system's one-shot learning capability for gestures. MHTrack's superior performance not only benefits from the inherent advantage of the siamese network but also stems from the stability of hand trajectory.

6) Impact of Scenarios: Different sensing environments face different level of background clutter such as multipath effect and thermal noise. To study the environmental robustness of MHTrack, we conduct experiments in two representative scenarios as follows.

Corridor: As a representative of common narrow scenarios such as balconies and alleys, the corridor has strong wall reflection interference. As shown in Fig. 23(a), the corridor is 2.77 m in height and 2.93 m in width. Points A and B are on a line

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.

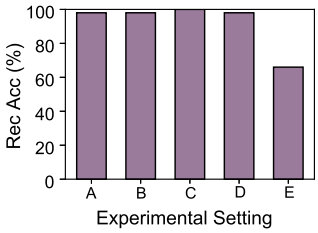


Fig. 24. Performance under different scenarios.

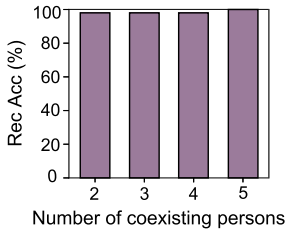


Fig. 25. Impact of coexisting persons.

directly in front of the radar, and two experimental settings are designed based on the points: (1) Standing at point A, which is 3.6 m away from the radar. (2) Walking towards the radar from point B 5.2 m away and executing gestures during the process. The experimental results are shown as A and B in Fig. 24.

The recognition accuracy of 98% on average indicates that MHTrack can work effectively in relatively narrow corridors.

Meeting room: Similar to complex scenarios like living rooms and bedrooms, the meeting room has many non-target reflectors, which cause complex multipath reflections that severely interfere with hand sensing. As shown in Fig. 23(b), the meeting room is 3.1 m in height and covers an area of $6.4\ \mathrm{m} \times 4.5$ m. As for interfering objects, the table is 0.75 m hith with dimensions $0.9\ \mathrm{m} \times 1.8$ m; the chairs have an average height of about 0.8 m and measure 0.48×0.48 m; and the display screen stands 1.95 m tall and is 1.7 m wide. We set up three locations C, D, and E, which are about 2m, 4.2m, and 4.8m away from the radar, respectively. There are three experimental settings: (1) Standing at Point C. (2) Walking towards the radar from point D and executing gestures during the process. (3) Standing at point E. The corresponding experimental results are shown as C, D and E in Fig. 24. The average accuracy under setting C and D is 99%, which indicates that the sensing environment with complex multi-path reflections does not have an influence on the recognition performance of MHTrack. Although the accuracy under setting E is only 66%, it is basically the same as the result in an empty scene (Section V-C1), which is caused by the long person-radar distance. Based on above results, we conclude that MHTrack has excellent environmental robustness.

7) Impact of Multi-Person Coexistence: It is a common situation in reality that multiple persons coexist in a scene. In general, the increase of person enhances the multipath reflection of signals, which may impact on hand tracking. Therefore, we evaluate the performance of MHTrack under different numbers of interfering subjects. When the target user performs a gesture, interfering individuals are present in the experimental area and exhibit various motion states such as standing, sitting, or walking. Additionally, considering that the multipath reflection from walking is stronger than in other states, we require that walking interfering persons be present throughout the entire experimental process, and they are allowed to be closer to the radar as long as they do not obscure the target user. Fig. 25 shows the gesture recognition accuracy with different numbers of presences.

TABLE IV
PERFORMANCE UNDER DIFFERENT LIGHTING CONDITIONS

Lighting Condition	Sunlight	Artificial Light	Darkness
Tracking Error	3.1cm	3cm	3.2cm
Recognition Accuracy	98%	100%	98%

The result shows that the gesture recognition accuracy remains above 96% when there are 2∼5 individuals present simultaneously in the scene, further demonstrating the robustness of MHTrack.

8) Impact of Gesture Category Number: Above experiments always evaluate 5 gestures. To further validate the one-shot learning capability of MHTrack, we add 25 new gestures to investigate whether the increase in gesture categories would affect recognition performance. The total 30 gestures include 10 symbols, 10 numbers and 10 letters, as shown in Fig. 29(a). Each newly added gesture is executed 11 times while the subject walking directly toward the radar, with one instance added to the database and the remaining 10 instances utilized for recognition testing. Together with the previous 5 kinds of gestures, a total of 300 samples are used to evaluate the recognition accuracy in this experiment. The confusion matrix of the recognition result is shown in Fig. 29(b). Based on the confusion matrix, the accuracy, precision, recall, and F1 score can be calculated for all gestures, and the average values are 99%, 99.1%, 99%, and 0.995, respectively. It can be seen that MHTrack has excellent extension capability for new gestures.

9) Impact of Lighting Condition: To verify whether MHTrack is light robust, we conduct experiments under three lighting conditions: sunlight, artificial light and darkness. For each condition, the subject stands in the center of sensing area, and we collect 10 trajectory samples (drawing circles in the air for 6 s per collection) to evaluate the tracking error and 50 gesture samples ($= 5$ gestures $\times$ 10 samples) to evaluate the recognition accuracy. The results are shown in Table IV, which indicate that our system is not affected by lighting conditions.

10) Effectiveness Verification of Anchor Correction: To verify the effectiveness of the anchor correction method in addressing gesture offset, this section conducts an ablation experiment using all samples from the five routes described in Section V-C4. First, the evaluation is conducted on the database with five default reference gestures, and the result is shown in Fig. 26. It can be observed that the average recognition accuracy of five route settings is 96.8% with correction and 95.6% without

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.

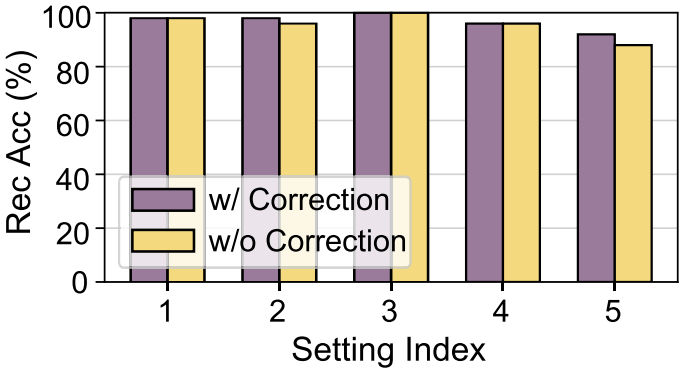


Fig. 26. Recognition accuracy under a 5-gestures database.

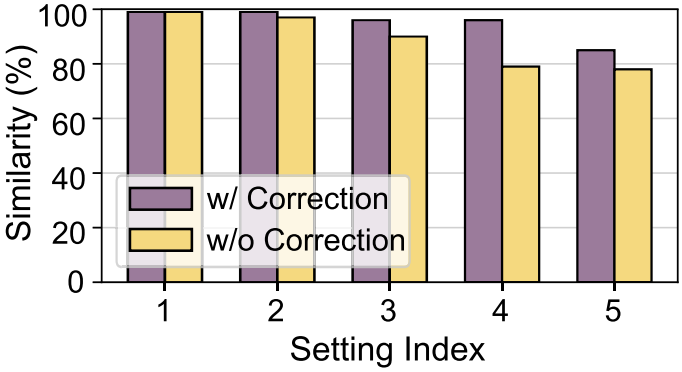


Fig. 27. Similarity of accurately recognized samples.

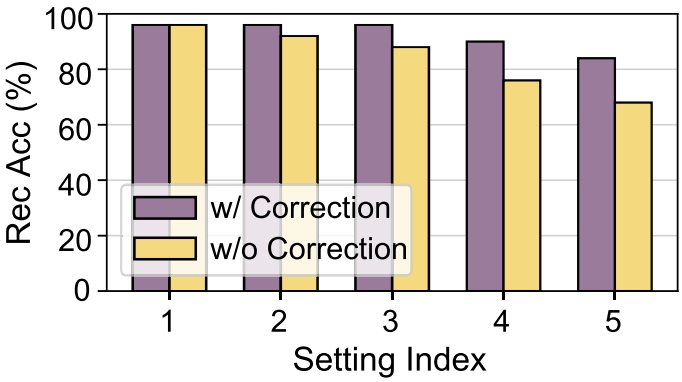
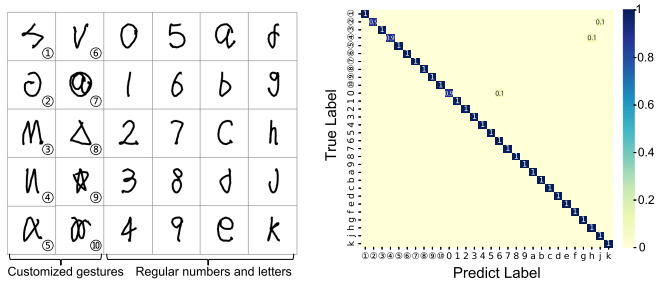


Fig. 28. Recognition accuracy under a 30-gestures database.

correction. Although the accuracy without correction decreases by only 1.2%, the similarity scores of correctly recognized samples drop significantly as the orientation shifts. As shown in Fig. 27, the similarity scores without correction on routes 4 and 5 are both below 80%, whereas the use of correction provides a higher similarity of 85% even on route 5. We deduce that the decent accuracy of without correction is due to the significant differences between reference gestures. If similar gestures exist in the gesture database, the absence of correction is likely to experience a substantial accuracy drop due to the presence of gesture offset. To verify this deduction, we repeat the experiment on the database with 30 reference gestures, and the result is shown in Fig. 28. It can be seen that the average



(a) Gesture visualization (b) Confusion matrix

Fig. 29. Impact of the number of gestures.

TABLE V
COMPUTATIONAL LATENCY

Module	Body Tracking	Hand Tracking	Gesture Recognition
Latency	2.50ms per frame	0.16s per frame	1.57s per gesture

accuracy with correction is 92.4%, while without correction, it drops to 84% and even as low as 68% on route 5. In conclusion, applying correction achieves superior performance compared to its absence, fully demonstrating the effectiveness of anchor correction in obtaining accurate relative hand trajectories.

11) Evaluation of Computational Latency: Finally, we evaluate the computational latency of MHTrack, which mainly consists of three parts: body tracking, hand tracking, and gesture recognition (the whole process from obtaining the gesture pattern to getting the final recognition result in the case of five gestures). The evaluation is performed on a desktop computer with an 11th Gen Intel(R) Core(TM) i7-11700 processor and 16 GB of RAM. The results are listed in Table V. Hand tracking accounts for the vast majority of the computational latency due to the need for re-angle estimation. Even so, MHTrack fulfills some degree of real-time requirements. We believe that specialized hardware acceleration could significantly increase the computational speed of this module.

VI. RELATED WORK

A. Hand Sensing Approaches

mmWave-based hand sensing generally includes two categories: hand tracking and gesture recognition. Regarding hand tracking, most studies limit hand motion to a small fixed area [14], [15], [31] or require more than one radar [32], [33], [34]. mmWrite [16] and AirWrite [17] extends the sensing range with small tracking errors, but its implementation requires the virtual writing panel and the radar to lie on the same plane, making it difficult to extend to hand tracking in 3D space. Regarding gesture recognition, many studies realize micro-gesture recognition at a close distance [25], [35], [36], [37], [38], [39], but they are insufficient for to meet the requirements of long-range applications such as remote control of home appliances. For this purpose, many studies aim at recognizing hand gesture by raw signal [40], [41], temporal spectrum [20], [42], [43], [44], [45], heatmap stream [1], [46], [47], [48], [49], [50] or point cloud [19], [21], [44], [51]. However, since the signal

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.

features are abstract and sensitive to the environment, the above methods often require extensive data acquisition and additional training in a new domain. Pantomime [3] decreases the impact of different positions by standardizing the point cloud location. DI-Gesture [18] proposes a data augmentation framework to reduce the data collection efforts. mTransSee [22] extracts the Doppler feature matrix and utilizes transfer learning to enhance the robustness of the system. Although the radar data collection issue is alleviated, these studies still have room for improvement in terms of the extensibility of the gesture. Moreover, existing works mainly focus on stationary body state, without supporting gesture interaction when the user is walking. For better user experience, this paper provides a novel hand tracking scheme that supports user movement, and further realizes easily extendable gesture recognition.

B. Other Sensing Approaches

Vital sign detection [52], [53], [54] and human body tracking [55], [56] are important representatives of micro-sensing and macro-sensing applications, which make significant research progress. Thanks to the high displacement sensitivity of millimeter waves, numerous studies emerge in the field of speech recognition [57], [58], [59], [60]. In recent years, the research on human mesh reconstruction and pose estimation [61], [62] make breakthroughs and raise concerns about user privacy [63], [64]. The improvement in sensing granularity enhances the ability to measure human emotions and object properties [65], [66], [67], [68], [69], [70], [71].

VII. Conclusion

The paper proposed the MHTrack system, which is the first to realize mmWave-based hand tracking for mobile users. MHTrack focuses on enabling robust and precise single-hand tracking in user mobility scenarios, and three major technical challenges were addressed. First, MHTrack solves problems of noise interference and unknown gesture execution timing through the wake-up detection and state transition mechanism. Second, to overcome the issue of hand point cloud disappearance, we proposed the local spatial update strategy to achieve stable hand tracking. Third, the proposed anchor correction and optimal panel search methods successfully solve the gesture offset problem. Experimental results revealed that MHTrack achieves high-precision hand tracking, and the absolute hand trajectory tracking error is only 3.6 cm in an area of $3.2\ \mathrm{m} \times 4.8\ \mathrm{m}$. For the case study of gesture recognition, we achieved the one-shot extension of new gestures by combining the siamese network, achieving recognition accuracy higher than 96% even with a 30-gesture classification. From another perspective, achieving excellent gesture recognition performance using a simple network further demonstrates the superior tracking capability of MHTrack. We believe that designing a more advanced network can enable even better recognition performance. As an extension of this work, we consider two-hand tracking to be an interesting and challenging scenario. In fact, the ideas behind MHTrack, such as wake-up detection, state transition, and local spatial update, are generalizable. For two-hand tracking, we need to carefully consider the correspondence between the body and hands. Moreover, the distance between two hands, especially when they cross and then separate, presents a significant challenge, which we leave for future exploration.

References

[1] E. Hayashi et al., "RadarNet: Efficient gesture recognition technique utilizing a miniature radar sensor," in *Proc. Annu. ACM Conf. Hum. Factors Comput. Syst.*, 2021, pp. 1–14.

[2] P. S. Santhalingam, A. A. Hosain, D. Zhang, P. Pathak, H. Rangwala, and R. Kushalnagar, "mmASL: Environment-independent ASL gesture recognition using 60 GHz millimeter-wave signals," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 4, no. 1, pp. 1–30, 2020.

[3] S. Palipana, D. Salami, L. A. Leiva, and S. Sigg, "Pantomime: Mid-air gesture recognition with sparse millimeter-wave radar point clouds," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 5, no. 1, pp. 1–27, 2021.

[4] P. Kang, J. Li, B. Fan, S. Jiang, and P. B. Shull, "Wrist-worn hand gesture recognition while walking via transfer learning," *IEEE J. Biomed. Health Inform.*, vol. 26, no. 3, pp. 952–961, Mar. 2022.

[5] J. Li et al., "Signring: Continuous american sign language recognition using IMU rings and virtual IMU data," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 7, no. 3, pp. 1–29, 2023.

[6] Y. Zhang, Y. Chen, H. Yu, X. Yang, R. Sun, and B. Zeng, "A feature adaptive learning method for high-density sEMG-based gesture recognition," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 5, no. 1, pp. 1–26, 2021.

[7] D. G. León et al., "Video hand gestures recognition using depth camera and lightweight CNN," *IEEE Sensors J.*, vol. 22, no. 14, pp. 14 610–14 619, Jul. 2022.

[8] W. Liu, H. Du, B. Li, J. Li, Z. Chang, and L. Wang, "RFPad: Enabling device-free handwriting recognition with a tag square," *IEEE Trans. Human-Mach. Syst.*, vol. 53, no. 2, pp. 325–334, Apr. 2023.

[9] X. Liu, D. Liu, J. Zhang, T. Gu, and K. Li, "RFID and camera fusion for recognition of human-object interactions," in *Proc. ACM Annu. Int. Conf. Mobile Comput. Netw.*, 2021, pp. 296–308.

[10] Z. Han, Z. Lu, Z. Hu, Y. Chen, and X. Wen, "On position-independency passive gesture tracking with commodity Wi-Fi," *IEEE Sensors J.*, vol. 23, no. 14, pp. 16 264–16 275, Jul. 2023.

[11] Z. Han, Z. Lu, X. Wen, L. Guo, and J. Zhao, "Towards 3D centimeter-level passive gesture tracking with two WiFi links," *IEEE Trans. Mobile Comput.*, vol. 22, no. 5, pp. 3031–3045, May 2023.

[12] C. Feng et al., "Wi-learner: Towards one-shot learning for cross-domain Wi-Fi based gesture recognition," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 6, no. 3, pp. 1–27, 2022.

[13] Y. Zhou, H. Chen, C. Huang, and Q. Zhang, "WiAdv: Practical and robust adversarial attack against WiFi-based gesture recognition system," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 6, no. 2, pp. 1–25, Jul. 2022.

[14] Y. Li, C. Gu, and J. Mao, "4-D gesture sensing using reconfigurable virtual array based on a 60-GHz FMCW MIMO radar sensor," *IEEE Trans. Microw. Theory Techn.*, vol. 70, no. 7, pp. 3652–3665, Jul. 2022.

[15] C. Yuan, Y. Zhong, J. Tian, and Y. Zou, "A real-time digit gesture recognition system based on mmWave radar," in *Proc. IEEE Int. Conf. Mach. Learn. Appl.*, 2022, pp. 770–775.

[16] S. D. Regani, C. Wu, B. Wang, M. Wu, and K. J. R. Liu, "mmWrite: Passive handwriting tracking using a single millimeter-wave radio," *IEEE Internet Things J.*, vol. 8, no. 17, pp. 13 291–13 305, Sep. 2021.

[17] C. Lin et al., "AirWrite: An aerial handwriting trajectory tracking and recognition system with mmWave," *IEEE Trans. Mobile Comput.*, vol. 23, no. 12, pp. 13 325–13 341, Dec. 2024.

[18] Y. Li et al., "Towards domain-independent and real-time gesture recognition using mmWave signal," *IEEE Trans. Mobile Comput.*, vol. 22, no. 12, pp. 7355–7369, Dec. 2023.

[19] H. Xie, P. Han, C. Li, Y. Chen, and S. Zeng, "Lightweight midrange arm-gesture recognition system from mmWave radar point clouds," *IEEE Sensors J.*, vol. 23, no. 2, pp. 1261–1270, Jan. 2023.

[20] J. Wu, J. Wang, Q. Gao, M. Cheng, M. Pan, and H. Zhang, "Toward robust device-free gesture recognition based on intrinsic spectrogram of mmWave signals," *IEEE Internet Things J.*, vol. 9, no. 19, pp. 19 318–19 329, Oct. 2022.

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.

[21] D. Salami, R. Hasibi, S. Palipana, P. Popovski, T. Michoel, and S. Sigg, "Tesla-rapture: A lightweight gesture recognition system from mmWave radar sparse point clouds," *IEEE Trans. Mobile Comput.*, vol. 22, no. 8, pp. 4946–4960, Aug. 2023.

[22] H. Liu et al., "Mtranssee: Enabling environment-independent mmWave sensing based gesture recognition via transfer learning," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 6, no. 1, pp. 1–28, 2022.

[23] H. Xue et al., "MmMesh: Towards 3D real-time dynamic human mesh construction using millimeter-wave," in *Proc. ACM Annu. Int. Conf. Mobile Syst. Appl. Serv.*, 2021, pp. 269–282.

[24] H. Kong et al., "M3Track: Mmwave-based multi-user 3D posture tracking," in *Proc. 20th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, 2022, pp. 491–503.

[25] H. Liu et al., "M-gesture: Person-independent real-time in-air gesture recognition using commodity millimeter wave radar," *IEEE Internet Things J.*, vol. 9, no. 5, pp. 3397–3415, Mar. 2022.

[26] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2019, pp. 6105–6114.

[27] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 539–546.

[28] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum, "Human-level concept learning through probabilistic program induction," *Science*, vol. 350, no. 6266, pp. 1332–1338, 2015.

[29] AWRx cascaded radar RF evaluation module (MMWCAS-RF-EVM), 2020. [Online]. Available: https://www.ti.com/lit/ug/swru553a/swru553a.pdf

[30] Nokov product data sheet (model pluto1.3c), 2023. [Online]. Available: https://en.nokov.com/colourpage/upload/Pluto1.3C_Data_Sheet231031.pdf

[31] W. Li, J. Jiang, Y. Y. Yao, D. Liu, Y. Gao, and Q. Li, "Hand character gesture recognition based on a single millimetre–wave radar chip," *IET Radar, Sonar Navigation*, vol. 16, pp. 208–223, 2022.

[32] M. Arsalan and A. Santra, "Character recognition in air-writing based on network of radars for human-machine interface," *IEEE Sensors J.*, vol. 19, no. 19, pp. 8855–8864, Oct. 2019.

[33] Y. Zhao, T. Liu, X. Feng, Z. Zhao, W. Cui, and Y. Fan, "New application: A hand air writing system based on radar dual view sequential feature fusion idea," *Remote Sens.*, vol. 14, no. 20, 2022, Art. no. 5177.

[34] M. Arsalan, T. Zheng, A. Santra, and V. Issakov, "Contactless low power air-writing based on FMCW radar networks using spiking neural networks," in *Proc. IEEE Int. Conf. Mach. Learn. Appl.*, 2022, pp. 931–935.

[35] H. Wei et al., "IndexPen: Two-finger text input with millimeter-wave radar," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 6, no. 2, pp. 1–39, 2022.

[36] C. Wang, X. Zhao, and Z. Li, "DCS-CTN: Subtle gesture recognition based on TD-CNN-transformer via millimeter-wave radar," *IEEE Internet Things J.*, vol. 10, no. 20, pp. 17 680–17 693, Oct. 2023.

[37] H. Lee, Y. Lee, H. Choi, and S. Lee, "Digit recognition in air-writing using single millimeter-wave band radar system," *IEEE Sensors J.*, vol. 22, no. 10, pp. 9387–9396, May 2022.

[38] W. Li, T. He, N. Jing, and L. Wang, "mmHSV: In-air handwritten signature verification via millimeter-wave radar," *ACM Trans. Internet Things*, vol. 4, no. 4, 2023, pp. 1–22.

[39] N. Hendy, H. M. Fayek, and A. Al-Hourani, "Deep learning approaches for air-writing using single UWB radar," *IEEE Sensors J.*, vol. 22, no. 12, pp. 11 989–12 001, Jun. 2022.

[40] B. Jin, Y. Peng, X. Kuang, Z. Zhang, Z. Lian, and B. Wang, "Robust dynamic hand gesture recognition based on millimeter wave radar using atten-TsNN," *IEEE Sensors J.*, vol. 22, no. 11, pp. 10 861–10 869, Jun. 2022.

[41] Z. Yu, D. Zhang, Z. Wang, Q. Han, B. Guo, and Q. Wang, "SoDar: Multitarget gesture recognition based on SIMO Doppler radar," *IEEE Trans. Human-Mach. Syst.*, vol. 52, no. 2, pp. 276–289, Apr. 2022.

[42] B. Jin, X. Ma, Z. Zhang, Z. Lian, and B. Wang, "Interference-robust millimeter-wave radar-based dynamic hand gesture recognition using 2D CNN-transformer networks," *IEEE Internet Things J.*, vol. 11, no. 2, pp. 2741–2752, Jan. 2024.

[43] S. Ahmed, W. Kim, J. Park, and S. H. Cho, "Radar-based air-writing gesture recognition using a novel multistream CNN approach," *IEEE Internet Things J.*, vol. 9, no. 23, pp. 23 869–23 880, Dec. 2022.

[44] Z. Xia and F. Xu, "Time-space dimension reduction of millimeter-wave radar point-clouds for smart-home hand-gesture recognition," *IEEE Sensors J.*, vol. 22, no. 5, pp. 4425–4437, Mar. 2022.

[45] J. Jung, S. Lim, J. Kim, and S.-C. Kim, "Digit recognition using FMCW and UWB radar sensors: A transfer learning approach," *IEEE Sensors J.*, vol. 23, no. 16, pp. 18 776–18 784, Aug. 2023.

[46] H. Zhao, Y. Ma, Y. Lu, and K. Liu, "DGSCR: Double-target gesture separation and classification recognition based on deep learning and millimeter-wave radar," *IEEE Sensors J.*, vol. 23, no. 21, pp. 26701–26711, Nov. 2023.

[47] X. Dong, Z. Zhao, Y. Wang, T. Zeng, J. Wang, and Y. Sui, "FMCW radar-based hand gesture recognition using spatiotemporal deformable and context-aware convolutional 5-D feature representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5107011.

[48] Y. Wang, D. Wang, Y. Fu, D. Yao, L. Xie, and M. Zhou, "Multi-hand gesture recognition using automotive FMCW radar sensor," *Remote Sens.*, vol. 14, no. 10, 2022, Art. no. 2374.

[49] Z. Liu, H. Liu, and C. Ma, "A robust hand gesture sensing and recognition based on dual-flow fusion with FMCW radar," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 4028105.

[50] C. Wang, T. S. Kumar, W. D. Raedt, G. Camps, H. Hallez, and B. Vanrumste, "Eat-radar: Continuous fine-grained eating gesture detection using FMCW radar and 3D temporal convolutional network," *IEEE J. Biomed. Health Inform.*, vol. 28, no. 2, pp. 1000–1011, Feb. 2024.

[51] J. Wang, Z. Ran, Q. Gao, X. Ma, M. Pan, and K. Xue, "Multi-person device-free gesture recognition using mmWave signals," *China Commun.*, vol. 18, pp. 186–199, 2021.

[52] C. Xu et al., "CardiacWave: A mmWave-based scheme of non-contact and high-definition heart activity computing," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 5, no. 3, pp. 1–26, 2021.

[53] U. M. Khan, L. Rigazio, and M. Shahzad, "Contactless monitoring of PPG using radar," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 6, no. 3, pp. 1–30, 2022.

[54] J. Gong, X. Zhang, K. Lin, J. Ren, Y. Zhang, and W. Qiu, "RF vital sign sensing under free body movement," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 5, no. 3, pp. 1–22, 2021.

[55] W. Chen et al., "Environment-aware multi-person tracking in indoor environments with mmWave radars," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 7, no. 3, pp. 1–29, 2023.

[56] D. Cao, R. Liu, H. Li, S. Wang, W. Jiang, and C. X. Lu, "Cross vision-RF gait re-identification with low-cost RGB-D cameras and mmWave radars," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 6, no. 3, pp. 1–25, 2022.

[57] J. Zhang, Y. Zhou, R. Xi, S. Li, J. Guo, and Y. He, "Ambiear: Mmwave based voice recognition in NLoS scenarios," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 6, no. 3, pp. 1–25, 2022.

[58] C. Wang, F. Lin, Z. Ba, F. Zhang, W. Xu, and K. Ren, "WavesDropper: Through-wall word detection of human speech via commercial mmwave devices," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 6, no. 2, pp. 1–26, 2022.

[59] S. Zeng, H. Wan, S. Shi, and W. Wang, "mSilent: Towards general corpus silent speech recognition using COTS mmWave radar," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 7, no. 1, pp. 1–28, 2023.

[60] R. Zhao, J. Yu, H. Zhao, and E. C. Ngai, "Radio2Text: Streaming speech recognition using mmWave radio signals," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 7, no. 3, pp. 1–28, 2023.

[61] A. Adhikari, H. Regmi, S. Sur, and S. Nelakuditi, "Mishape: Accurate human silhouettes and body joints from commodity millimeter-wave devices," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 6, no. 3, pp. 1–31, 2022.

[62] H. Ding et al., "MI-mesh: 3D human mesh construction by fusing image and millimeter wave," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 7, no. 1, pp. 1–24, 2023.

[63] H. Liu, X. Liu, X. Xie, X. Tong, T. Shi, and K. Li, "Application-oriented privacy filter for mmWave radar," *IEEE Commun. Mag.*, vol. 61, no. 12, pp. 168–174, Dec. 2023.

[64] J. Shenoy, Z. Liu, B. Tao, Z. Kabelac, and D. Vasisht, "RF-protect: Privacy against device-free human tracking," in *Proc. ACM SIGCOMM Comput. Commun. Rev.*, 2022, pp. 588–600.

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.

[65] X. Zhang, Y. Zhang, Z. Shi, and T. Gu, "MmFER: Millimetre-wave radar based facial expression recognition for multimedia IoT applications," in *Proc. ACM Annu. Int. Conf. Mobile Comput. Netw.*, 2023, pp. 1–15.
[66] K. Liang, A. Zhou, Z. Zhang, H. Zhou, H. Ma, and C. Wu, "mmStress: Distilling human stress from daily activities via contact-less millimeter-wave sensing," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 7, no. 3, pp. 1–36, 2023.
[67] Y. Yang, H. Xu, Q. Chen, J. Cao, and Y. Wang, "Multi-vib: Precise multi-point vibration monitoring using mmWave radar," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 6, no. 4, pp. 1–26, 2023.
[68] Y. Han et al., "Enhancing non-contact vibration monitoring with mmWave radar and camera fusion," *IEEE Internet Things J.*, vol. 12, no. 7, pp. 7888–7901, Apr. 2025.
[69] C. Wu, F. Zhang, B. Wang, and K. J. R. Liu, "mSense: Towards mobile material sensing with a single millimeter-wave radio," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 4, no. 3, pp. 1–20, 2020.
[70] Y. Liang, A. Zhou, H. Zhang, X. Wen, and H. Ma, "FG-liquiD: A contact-less fine-grained liquid identifier by pushing the limits of millimeter-wave sensing," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 5, no. 3, pp. 1–27, 2021.
[71] M. Xue, Y. Chen, X. Gong, J. Zhang, and C. Fan, "Wet-Ra: Monitoring diapers wetness with wireless signals," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 6, no. 2, pp. 1–26, 2022.



Xiulong Liu (Senior Member, IEEE) received the BE and PhD degrees from the Dalian University of Technology (China) in 2010 and 2016, respectively. He is currently a professor with the College of Intelligence and Computing, Tianjin University, China. He also worked as a visiting researcher in Aizu University, Japan; a postdoctoral fellow in The Hong Kong Polytechnic University, Hong Kong; and a postdoctoral fellow with the School of Computing Science, Simon Fraser University, Canada. His research interests include wireless sensing and communication, indoor localization, and networking, etc. His research papers were published in many prestigious journals and conferences including *IEEE/ACM Transactions on Networking*, *IEEE Transactions on Mobile Computing*, *IEEE Transactions on Computers*, *IEEE Transactions on Parallel and Distributed Systems*, TCOM, INFOCOM, and ICNP, etc. He received Best Paper Awards from ICA3PP 2014 and IEEE System Journal 2017. He is also the recipient of CCF Outstanding Doctoral Dissertation award 2017.



Hankai Liu received the BE degree in Internet of Things engineering and the MS degree in electronic and communication engineering from Tianjin University, China, in 2018 and 2021, respectively, where he is currently working toward the PhD degree in computer science and technology. His current research interests include mmWave radar and RF computing.



Yantao Han received the BE degree in software engineering from Zhengzhou University, China, in 2022. He is currently working toward the MS degree in Tianjin University, Tianjin, China. His current research interests include mmWave radar and wireless sensing.



Xin Xie (Member, IEEE) received the BE and PhD degrees from the School of Computer Science and Technology, Dalian University of Technology, China, in 2013 and 2019, respectively. He is currently a professor with the College of Intelligence and Computing, Tianjin University, China. He also worked as a visiting scholar with the Department of Computer Sciences, Purdue University, USA, a postdoctoral fellow with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong SAR, and a researcher with the Theory Laboratory of 2012 Labs, Huawei, Hong Kong SAR. His research interests include IoT, edge intelligence, and network systems.



Xinyu Tong received the BE and PhD degrees from the Department of Electronic Information and Electrical Engineering from Shanghai Jiao Tong University, Shanghai, China in 2015 and 2020, respectively. He is currently an associate researcher with the College of Intelligence and Computing of Tianjin University. His research interests include wireless sensor network and wireless localization. His research papers were published in many prestigious journals and conferences including *IEEE Transactions on Mobile Computing*, *IEEE/ACM Transactions on Networking*, MobiCom, UbiComp, and INFOCOM, etc.



Keqiu Li (Fellow, IEEE) received the bachelor's and master's degrees from the Department of Applied Mathematics, the Dalian University of Technology in 1994 and 1997, respectively, and the PhD degree from the Graduate School of Information Science, Japan Advanced Institute of Science and Technology in 2005. He is currently a full professor, the dean of the College of Intelligence and Computing, Tianjin University, China. He is the recipient of National Science Foundation for Distinguished Young Scholars of China. He keeps working on the topics of blockchain, mobile computing, datacenter, and cloud computing. He has more than 150 papers published on prestigious journals or conferences, such as *IEEE/ACM Transactions on Networking*, *IEEE Transactions on Parallel and Distributed Systems*, *IEEE Transactions on Computers*, *IEEE Transactions on Mobile Computing*, INFOCOM, ICNP, etc.

Authorized licensed use limited to: NANKAI UNIVERSITY. Downloaded on October 31,2026 at 06:40:10 UTC from IEEE Xplore. Restrictions apply.