This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3066507, IEEE Internet of Things Journal

mmWrite: Passive Handwriting Tracking Using a Single Millimeter Wave Radio

Sai Deepika Regani, Chenshu Wu, Beibei Wang, Min Wu and K. J. Ray Liu

***Abstract*—In the era of pervasively connected and sensed Internet of Things, many of our interactions with machines have been shifted from conventional computer keyboards and mouses to hand gestures and writing in the air. While gesture recognition and handwriting recognition have been well studied, many new methods are being investigated to enable pervasive handwriting tracking. Most of the existing handwriting tracking systems either require cameras, handheld sensors or involve dedicated hardware restricting user-convenience and the scale of usage. In this paper, we present *mmWrite*, the first high-precision passive handwriting tracking system using a single commodity millimeter wave (mmWave) radio. Leveraging the short wavelength and large bandwidth of 60 GHz signals and the radar-like capabilities enabled by the large phased array, *mmWrite* transforms any flat region into an interactive writing surface that supports handwriting tracking at millimeter accuracy. *MmWrite* employs an end-to-end pipeline of signal processing to enhance the range and spatial resolution limited by the hardware, boost the coverage, and suppress interference from backgrounds and irrelevant objects. We implement and evaluate *mmWrite* on a commodity 60 GHz device. Experimental results show that *mmWrite* can track a finger/pen with a median error of 2.8 mm and thus can reproduce handwritten characters as small as 1 cm $\times$ 1 cm, with a coverage of up to 8 m^2 supported. With minimal infrastructure needed, *mmWrite* promises ubiquitous handwriting tracking for new applications in the field of human-computer interactions.**

***Index Terms*—Handwriting tracking, mmWave, 60 GHz, Radar, Passive tracking, HCI, WiFi sensing**

I. Introduction

Rapid increase in automation has stimulated the quest for more efficient and convenient approaches for Human-Computer Interaction (HCI). Touch screens and smart surfaces (e.g., electronic whiteboards) have emerged as a more user-friendly alternative to the traditional input devices such as the keyboard and the computer mouse. However, the smart surfaces are usually in small form factors limiting the space for HCI and hindering our vision of ubiquitous smart environments. Instead, we see an increasing trend of ubiquitous interactions with machines via in-the-air gestures, handwriting, and voice-controllable systems, evidenced by recent industrial efforts, including Google Soli [1], Apple UWB radar, Apple Siri, Amazon Alexa, Google Home, and so on. Handwriting is a convenient mode of interaction for people and can be considered as a more general form of a gesture, which is a promising approach for HCI. Aided by the advanced handwriting recognition systems [2]–[4], enabling robust and accurate handwriting tracking can realize countless applications in the field of HCI.

D. Regani, C. Wu, B. Wang, M. Wu and K. J. R. Liu are with the Department of Electrical and Computer Engineering, University of Maryland at College Park, College Park, MD 20742 USA, and also with Origin Wireless, Inc., Greenbelt, MD 20770 USA (e-mail: rdeepika@terpmail.umd.edu; {cswu, bebewang, minwu, kjrliu}@umd.edu).

Fig. 1: Illustration of handwriting tracking using the proposed *mmWrite* system. (a) Picture of the handwritten trajectory and (b) Reconstructed trajectory of the word “mmWrite”.

Handwriting tracking involves tracking and recovering the trajectory traced by a writing target (e.g., a finger, stylus pen or a marker) and can be achieved either actively or passively. Active systems need to instrument the moving target with sensors like smartphone accelerometers [5], RFID tags [6], or other radios. Differently, in passive tracking, there are no electronic devices attached to the target. Camera-based approaches [7], [8] dominate the passive tracking systems, yet impose restrictions on the availability of ambient light and raise privacy concerns. Other modalities, such as acoustic signals, have been used to differentiate patterns of hand movements [9], [10]. However, the performance is severely degraded with distance, and they usually need retraining for every new alphabet and surface [11].

Wireless signals, particularly WiFi signals, have been exploited to achieve pervasive device-free sensing for various applications [12]–[16], such as motion detection, vital sign monitoring, human identification, gait recognition, indoor tracking, etc. [17]–[21]. The capabilities of wireless sensing are fundamentally limited by parameters such as the bandwidth, wavelength, and the antenna amount, etc. While the bandwidth limits the range resolution that can be achieved, the wavelength affects the scale of changes that can be perceived. For example, the 802.11 ac WiFi operating on 2.4 GHz or 5 GHz band with bandwidths up to 80 MHz offers about a range resolution of several meters, which is much lower than that required for handwriting tracking. Recently, two trends have arisen that may underpin much finer-resolution wireless sensing, including high-precision handwriting tracking. First, 60GHz millimeter-wave WiFi, introduced for high-rate networking with 802.11ad/ay standards, brings distinct advantages for wireless sensing [22], [23]. Compared to 2.4 GHz/5 GHz WiFi signals, mmWave signals offer shorter wavelengths that create stronger reflections off small objects and enable motion capturing at millimeter scales. Also, mmWave radios usually have bigger phased antenna array, producing higher directivity

and fewer multipaths that allows better spatial resolution. 60GHz WiFi has been offered in commercial routers and is being put into smartphones [24]. Second, while wireless sensing aims to enable radar-like features using radio signals proposed for communication, indoor portable radars are also becoming popular. FMCW has been widely exploited for wireless and acoustic sensing [25]–[27], while UWB radar and mmWave radar have been equipped into commodity smartphones. Inspired by these two trends, the joint design of communication and sensing systems emerges as an attractive direction [28].

In this work, we leverage a synergy between the above two trends by reusing a commodity 60GHz networking device as a mmWave radar. We present *mmWrite*, the first passive motion tracking system that can recover the trajectory of handwritten traces with high precision using a single commodity 60GHz mmWave radio. *mmWrite* captures the signals reflected off a writing object (e.g., a finger or a pen) and employs a pipeline of signal processing to reproduce the moving trajectory. And unlike gesture recognition systems, *mmWrite* does not require any calibration or supervised learning. Instead, it directly tracks the location of the writing target and recovers the entire trace. With one single radio and without instrumenting the writing object, *mmWrite* achieves handwriting tracking with millimeter accuracy over a large area of up to 8 m^2, thus converting a conventional surface into an interactive trackpad with minimal supporting infrastructure. We have shown one instance of handwriting tracking in Fig. 1.

Utilizing a mmWave radio to deliver such a handwriting tracking system, however, entails unique challenges in practice. First, the 60GHz mmWave signals, although with a large bandwidth of a few GHz, does not offer adequate range resolution for mm-scale handwriting tracking. In our test bed, the range resolution is 4.26 cm given the bandwidth of 3.52 GHz, far worse than the desired tracking accuracy at millimeter level. Second, given the high attenuation nature and near-far effects of mmWave signals (i.e., the spatial resolution of the directional signals decreases over distance), it is non-trivial to achieve a large coverage using a single radio. Third, reflection signals from background environments and especially irrelevant moving objects (e.g., body and arms) can distort the reflections from the target of interest (e.g., the finger) and thus influence the tracking performance.

The mmWrite algorithm is designed to conquer the above challenges and build a robust handwriting tracking system. For this, the time series of the channel impulse response (CIR) is captured during handwriting. The time-of-arrival (ToA) information embedded in the CIR, along with the spatial information extracted by digital beamforming, contains a rough estimate of the received signals' spatial origin. The contribution of the irrelevant signals from the environment to the received signal is suppressed by background subtraction. Moreover, the Doppler spectrum is used to isolate the signals reflected from the dynamic targets. The presence of the target is then detected from the Doppler spectrum using the Constant False Alarm Rate (CFAR) technique, and the corresponding spatial location is extracted. The coarse spatial information of the target is then refined further using the subsample peak interpolation technique. These target localization steps are repeated for each time instance to construct an estimated trajectory of the target, which is then enhanced to output the final trajectory using Discrete Cosine Transform-based smoothing technique.

We implement *mmWrite* on a commodity 60GHz 802.11ad device sponsored by Qualcomm, which is turned into a radar-like mode by attaching an extra antenna array to the radio. The performance of *mmWrite* is validated using three different approaches. First, we visually examine the shape-preserving capability of *mmWrite* by showing the recovered trajectories, which demonstrate that *mmWrite* can reproduce characters spanning a few centimeters that are recognizable by human eyes. We also feed the recovered trajectories to a standard handwriting recognition software, which reports an accuracy of 80% for 3 cm × 3 cm characters written at a distance of 20 cm. Finally, we quantify the tracking error at different ranges and azimuth angles. *mmWrite* achieves a median tracking error of 2.8 mm at a range of 20 cm from the device and can track handwriting in a region spanning about 8 m^2.

The main contributions of *mmWrite* are as follows:

- We present *mmWrite*, the first handwriting tracking system by reusing a single commodity 60GHz WiFi radio as a mmWave radar. With minimal infrastructure support, *mmWave* can transform any flat surface, be it a conventional whiteboard, a table, or a wall, into an interactive writing area.
- We present a complete pipeline to track a writing object at millimeter accuracy and thus retrieve handwritten characters as small as 1 cm x 1 cm over a coverage up to about 8 m^2 and a maximum range of 3 m.
- We perform numerous experiments to validate the performance of *mmWrite* in different conditions and environments, including varying distances, surfaces, writing speeds, and device parameters.

This paper is organized as follows. Preliminary information required to understand the current work is given in Section II. Section III discusses various stages of *mmWrite* in detail, followed by Section IV, which evaluates the performance of *mmWrite* through different experiments. We review the relevant literature in Section V and present the future work in Section VI. Finally, we conclude the paper in section VII.

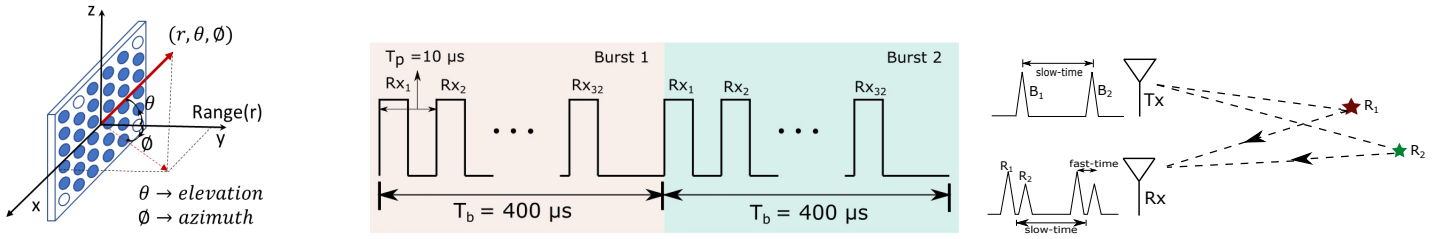
II. Preliminaries

A. The 60GHZ radar

Our experimental testbed is provided by Qualcomm, which reuses a commodity 60GHz 802.11ad chipset as a pulsed mmWave radar by attaching an additional antenna array to the radio. The co-located Tx and Rx arrays are both equipped with 32 elements, arranged in a 6×6 grid with a separation of 3 mm between the adjacent pair of antennas. The arrangement of the 32 antennas and the coordinate system of the radar is shown in Fig. 2a.

The pulsed radar transmits electromagnetic waves as discrete pulses reflected by various objects and received by the receiver antenna. A burst is a group of 32 pulses and is transmitted by one Tx. Each of the 32 received pulses

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3066507, IEEE Internet of Things Journal



(a) Radar coordinate system. (b) Concept of burst and pulse in radar (c) Concept of slow-time and fast-time

Fig. 2: Basic radar concepts used in *mmWrite*.

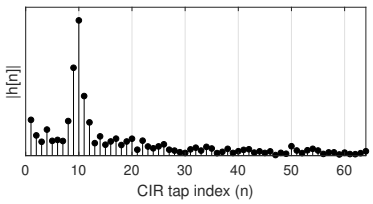


Fig. 3: Channel Impulse Response.

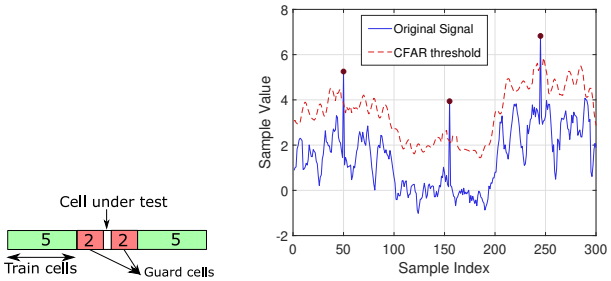
is detected by the receiving antenna sequentially, and the corresponding CIR is recorded. Fig. 2b shows two bursts each consisting of 32 pulses.The CIR is computed in the hardware sequentially by switching the antenna elements and uploaded to MATLAB. This CIR is used as the input to the *mmWrite* pipeline.

In this work, two different notions of time will be used to extract information about the writing object. The burst dimension is known as the *slow-time* dimension while the time dimension corresponding to the CIR taps is known as the *fast-time*. The concept of *slow-time* and *fast-time* is demonstrated in Fig. 2c. Consider one Tx, one Rx and two reflectors R_1 and R_2. Assume the Tx transmits two bursts, B_1 and B_2, which are reflected and received by the Rx. Here a burst consists of only one pulse since we assume only one Rx. For each burst, the two reflections from R_1 and R_2 result in different time-of-arrival (ToA) due to different path lengths which are captured on different CIR taps. For instance, Fig. 3 shows one realization of CIR, **h** consisting of 64 taps. A bandwidth of BW = 3.52 GHz[1] on our mmWave platform allows a time resolution of 0.28 ns (Δt), i.e., signals whose propagation delay differ by greater than Δt are recorded on different taps. The CIR tap index n, can therefore give an estimate of the range of the reflector. On the other hand, the *slow-time* captures the temporal difference in the position of a moving target which will be used to differentiate the moving target from the static objects. This will be elaborated more in §III. A time resolution of 0.28 ns translates to a range accuracy of $\frac{c\Delta t}{2}$ = 4.261 cm, where c is the speed of light. The antenna response of the antenna array is measured and available on a grid on 2 degrees in the azimuth and elevation dimensions and hence accuracy of 2 degrees can be achieved in those dimensions.

[1]Each channel in the 60GHz band of 802.11 ad spans a bandwidth of 2.16 GHz, of which 1.76 GHz is useful. A bandwidth of 3.52 GHz is achieved by channel bonding of two channels.

B. Target Detection

In this work, we use the CA-CFAR (Cell Averaging-Constant False Alarm Rate) technique for target detection. Fig. 4 demonstrates the CFAR technique for a one-dimensional signal. Consider a noisy signal with three targets of interest indicated by dots in Fig. 4b. The CFAR window is designed as shown in Fig. 4a with the guard cells as zeros and train cells as ones. This CFAR window is convolved with the signal to estimate the average noise level. An additional fixed threshold (equal to 2 in this example) is added to the estimated noise level to obtain the adaptive CFAR threshold. The locations at which the signal is greater than the CFAR threshold are identified as target locations.



(a) (b)

Fig. 4: CA-CFAR for one dimensional signal.

C. Subsample Interpolation

Subsample Peak Interpolation (SPI) is used to determine the location of peak of a signal at sub-sample accuracy in different applications [29], [30]. To include the information from the adjacent taps, we use the parabolic interpolation technique, which is demonstrated in Fig. 5. The circular points correspond to the discrete signal around a potential local maximum. By inspection, the center point (X=0) is considered as the location of the peak. However, using a parabolic fit through the three points helps us to estimate the location of the peak at sub-sample accuracy. As we will see in §III-E, SPI provides a significant improvement in the resolution of the target location.

III. mmWrite Algorithm

This section describes the handwriting tracking algorithm of *mmWrite* in detail. The algorithm aims to recover the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3066507, IEEE Internet of Things Journal

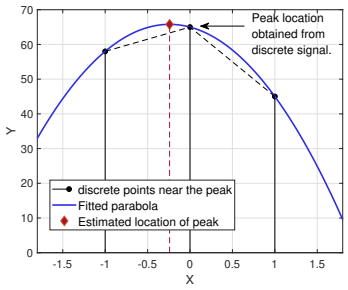


Fig. 5: Subsample peak interpolation.

handwriting trajectory from the CIR time series recorded by the radar while writing. The CIR time series is processed in different modules, which are shown in Fig. 6. To begin with, background subtraction is performed to reduce the contribution of the static objects in the environment to the received signal. Next, digital beamforming is used to obtain the spatial information followed by the transformation to the Doppler domain, whose aim is to differentiate the moving/writing object from other static objects in the environment. The target is detected using the 3D-CFAR and clutter mapping techniques, which is further finely localized within the detected spatial bins using the subsample interpolation technique. In the final step, the trajectory points corresponding to the target of interest are gathered and combined to form a raw trajectory, further smoothed using the DCT-based smoothing technique to obtain the *mmWrite* handwriting trajectory.

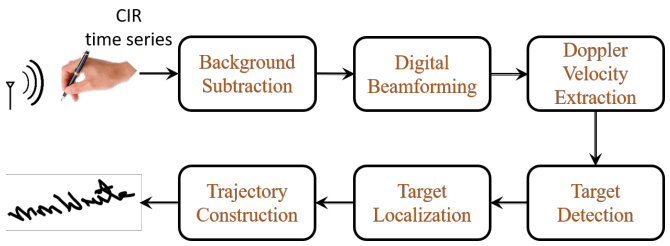


Fig. 6: Overview of *mmWrite*.

A. Background subtraction

Let the CIR recorded by the r^{th} Rx at time instance t be denoted by $h[r,n,t]$, where n is the tap index of the CIR. To highlight the target of interest (i.e., the writing object) and reduce the contribution from other static objects in the environment, the background CIR is subtracted from $\mathbf{h}$. In this work, the background CIR is estimated by taking the average of the CIR time series along the *slow-time* dimension. However, when designing a causal system or if the duration of the handwriting is short, a calibration step can be introduced, and the background can be recorded without the writing object. The background subtraction can be mathematically written as:

$$\hat{h}[r,n,t] = h[r,n,t] - \overline{h[r,n,.]}, \tag{1}$$

where $\overline{\mathbf{h}}$ is the average CIR over time and $\hat{\mathbf{h}}$ is the CIR after background subtraction. In the following steps, the notation $\mathbf{h}$ will be used instead of $\hat{\mathbf{h}}$ to denote the CIR after background subtraction. The tap index n corresponds to the *fast-time* and contains information about the range of the reflecting object. However, there could be multiple signals from different directions at the same range, making the target identification task difficult. To address this, the received signal is spatially filtered, thereby extracting the directional information by leveraging the antenna's phased array structure. Digital beamforming thus forms the next stage of the *mmWrite* algorithm.

B. Digital Beamforming

As we have the raw CIR at each receiving element, we can perform digital beamforming to separate the target-reflected signals from other irrelevant reflection signals. In this work, classical beamforming is used [31]. The steering vector used for beamforming was measured by Qualcomm and is available with the device. The Fraunhofer distance, which determines the near-field limit, is calculated as 18 cm. Hence, in this work, a range greater than 20 cm can be considered a far-field.

Let K reflected signals be incident on the receiving antenna array and let $\mathbf{g}_k$ denote the k^{th} incoming reflected signal. Let the azimuth and elevation angles corresponding to $\mathbf{g}_k$ be represented by az_k and el_k respectively. If the measured antenna response of the receiver r is denoted by $\mathbf{a}_r$, then the combined received signal $\mathbf{h}$ at the receiver antenna can be modeled as:

$$h[r,n,t] = \sum_{k=1}^{K} a_r(az_k, el_k) g_k[az_k, el_k, n, t], \tag{2}$$

where n is the CIR tap index and t is the time instance. The beamformed CIR, $\mathbf{y}$ is a four-dimensional matrix obtained by compensating for the array response as:

$$y[az, el, n, t] = \mathbf{a}^H(az, el) h(., n, t), \tag{3}$$

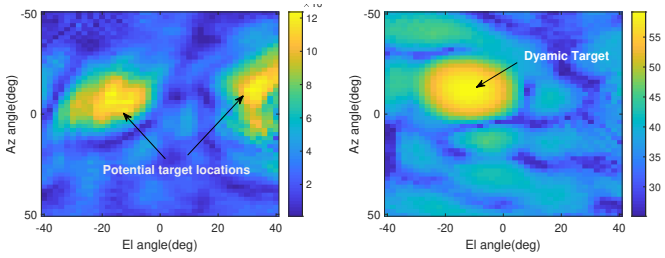
where $\mathbf{a}(az, el)$ is the 32-dimensional vector of antenna responses of all the receiver antennas for a signal arriving at an angle (az,el) and $(.)^{\mathbf{H}}$ is the Hermitian operator. To simplify the notation, a vector $\mathbf{T}$ is used to denote the triplet (az,el,n) as follows:

$$h[r,n,t] \overset{beamforming}{\longrightarrow} y[\underbrace{az, el, n}_{\mathbf{T}=[az,el,n]}, t] = y[\mathbf{T}, t]. \tag{4}$$

One instance of the beamformed CIR is shown in Fig. 7a. The variation of the absolute value of $\mathbf{y}$ at a range tap of 8, corresponding to a range of 34 cm, is shown with different azimuth and elevation angles.

At this stage, the range and direction of the reflected signals can be determined from the beamformed received signal. To differentiate the writing object from other static objects in the environment, *mmWrite* exploits its dynamic nature. Next, the beamformed CIR is further processed to extract the velocity information of the targets.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3066507, IEEE Internet of Things Journal



(a) (b)

Fig. 7: (a) Magnitude of the beamformed CIR, and (b) The maximum Doppler power matrix, $\mathbf{P}$, corresponding to the 8$^{\text{th}}$ CIR tap at one time instance, s. The matrix $\mathbf{P}$ retains the power contributed only by the dynamic targets. This resulted in suppression of one of the two potential targets observed in (a).

C. Doppler Velocity Extraction

The correspondence between the relative radial velocity of the target w.r.t the source and the observed frequency of the received signal, from the well-known Doppler effect, motivated us to transform the beamformed signal into the frequency domain. This transformation is achieved by performing a Short-Time Fourier Transform (STFT) along the *slow-time* dimension t. The Fourier transform applied in a window of length w_{l} and a step size of w_{s}. A smaller value of w_{s} increases the sampling frequency of points on the trajectory, reducing computational complexity. As w_s is halved, the complexity doubles. As we have more points on the trajectory, it can capture more drastic changes in the writing. More discussion is provided in §IV-B. Increasing w_{l} improves the resolution in the frequency domain; however, the instantaneous information is smoothed out. A value of $w_{\text{l}} = 192$ and $w_{\text{s}} = 32$ is used in this work unless mentioned otherwise. The transformation can be written as:

$$Y[\mathbf{T}, s, f] \overset{FFT}{\longleftarrow} y[\mathbf{T}, t+(s-1)w_{\text{s}} - w_{\text{l}} + 1 : t+(s-1)w_{\text{s}}], \quad (5)$$

where s is the window index which is also indicative of the *slow-time*. As the target moves, the observed frequency of the received signal is altered due to the Doppler effect. This change in frequency (Δf) depends on the direction and speed of the moving target and is given by:

$$\Delta f = \frac{2v}{c} f_{\text{c}}, \quad (6)$$

where v is the relative radial velocity of the target, f_{c} is the frequency of the carrier wave and c is the speed of light. This correspondence between the observed frequency, $f = f_c + \Delta f$, and the target radial velocity can be shown on the transformed CIR as:

$$Y[\mathbf{T}, s, f] \longleftrightarrow Y[\mathbf{T}, s, v]. \quad (7)$$

The reflected signals from the static objects in the environment contribute to the zero frequency bin of the transformed CIR, while those reflected from the moving targets contribute to the non-zero frequency bins corresponding to their relative radial velocity. Thus, to detect the moving target, i.e., the writing object, the power in the non-zero frequency bins must be analyzed. Among the non-zero frequency bins, the maximum Doppler power ($\mathbf{P}$) is extracted for each spatial bin (azimuth, elevation and range) and time instance as follows:

$$v^*_{\mathbf{T},s} = \underset{v \neq 0}{\text{argmax}} |Y[\mathbf{T}, s, v]|, \quad (8)$$

$$P[\mathbf{T}, s] = |Y[\mathbf{T}, s, v^*_{\mathbf{T},s}]|. \quad (9)$$

Recall that the non-zero frequency bins of the Doppler power capture the reflected signals corresponding to the target with non-zero radial velocity. However, during handwriting, there will be instances when the writing object has a zero radial velocity, and $\mathbf{P}$ does not capture the contribution from signals of interest. The zero radial velocity could be due to one of the following reasons: (a) The speed of the target is zero, or (b) the radial component of the velocity of the target is zero, i.e., the target is moving perpendicular to the range dimension. In such instances, the Doppler power corresponding to the zero frequency bin ($\mathbf{P}_0$) is analyzed to capture the reflected signals from the target of interest. Mathematically, this is written as,

$$P_0[\mathbf{T}, s] = |Y[\mathbf{T}, s, 0]|. \quad (10)$$

Fig. 7b shows an instance of the matrix $\mathbf{P}$ at a range tap of 8 for different azimuth and elevation angles. The four-dimensional matrices $\mathbf{P}$ and $\mathbf{P}_0$ indicate the Doppler power for the range dimension with a resolution of 4.26 cm, and azimuth and elevation dimensions with a spacing of 2 degrees for each time instance, s. These matrices are analyzed to detect and localize the target.

D. Target Detection

The spatial bins consisting of dynamic targets have a higher Doppler power compared to the ones without a dynamic target. Therefore, the goal of target detection is to identify the bins with relatively higher Doppler power to obtain a rough estimate of the location of the dynamic target. For this, the Doppler power values need to be compared with a threshold. However, the spatial and temporal variation in the noise level prevents us from using a universal threshold for target detection. This problem has been addressed in the radar literature with an adaptive threshold. The most commonly used technique is the CA-CFAR target detection [32].

In this work, a three-dimensional CFAR window is used, corresponding to the range, azimuth, and elevation dimensions. A CFAR threshold map is computed for $P[\mathbf{T}, s]$, and the spatial bins with Doppler power above the CFAR threshold are extracted. Let the 3D CFAR window be denoted by $\mathbf{C}_{\text{win}}$ and C_{add} be the additional threshold. The overall CFAR threshold matrix C^{th} is given by:

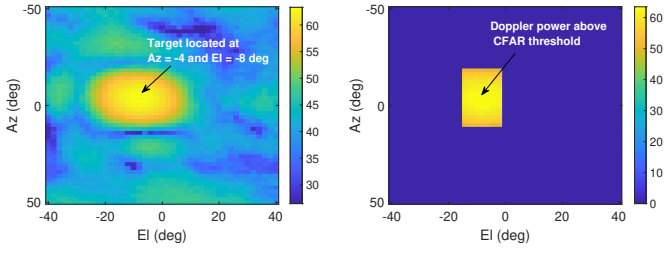
$$\mathbf{C}^{\text{th}} = \mathbf{C}_{\text{win}} \star \mathbf{P}[., s] + C_{\text{add}}, \quad (11)$$

where $\star$ is the convolution operation. The Doppler power matrix after applying the CFAR threshold, $\mathbf{P}^c$ is written as:

$$P^c[\mathbf{T}, s] = \begin{cases} P[\mathbf{T}, s], & \text{if } P[\mathbf{T}, s] > C^{\text{th}}[\mathbf{T}] \\ 0, & \text{otherwise.} \end{cases} \quad (12)$$

Fig. 8a shows the Doppler power for different azimuth and elevation angles and a fixed range tap of 7. The resulted

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3066507, IEEE Internet of Things Journal



(a) (b)

Fig. 8: (a) Doppler power at range tap 7 for different azimuth and elevation angles, (b) Doppler power after applying CFAR threshold. In this example, the target is detected at an elevation of -8° and an azimuth of -4° with a Doppler power of 63.5 dB.

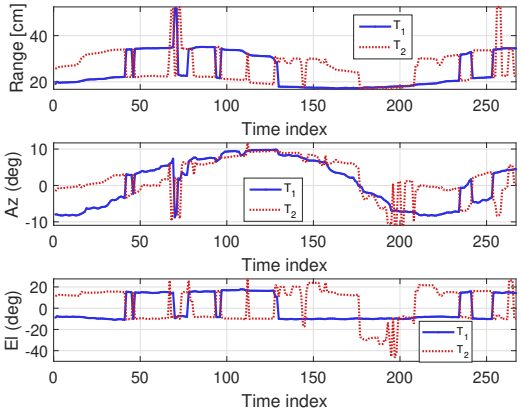


Fig. 9: Estimated target locations with time using the two highest Doppler power bins. The traces are marked as T_1 and T_2, corresponding to the first and second highest Doppler power respectively.

Doppler power after applying CFAR threshold is shown in Fig. 8b. Before designing the target detection module using the 3D-CFAR technique, let us understand the two challenges in *mmWrite*: Multiple moving parts and zero radial velocity.

Multiple moving parts: During handwriting, at any given instance of time, different parts of the hand (fingers, wrist, arm) move in different directions and speeds, resulting in different radial velocities due to different inclinations. As a result, the highest Doppler power does not necessarily correspond to the target of interest (the tip of the pen, for example) at all times. For instance, Fig. 9 shows the range, azimuth, and elevation corresponding to the two highest values in the Doppler power matrix $\mathbf{P}$ with time. The traces T_1 and T_2 indicate the spatial information corresponding to the first and second highest Doppler power bins. As shown, two distinct target traces can be observed, and the highest Doppler power keeps switching between the two. The Doppler power at a particular spatial bin and time depends on many factors such as the instantaneous radial velocity of the dynamic target if present, the radar cross-section, material, and location of the target. To avoid misdetection of the target of interest, N targets are detected at each instance instead of one. These N targets are identified iteratively in the decreasing order of Doppler power by comparing with the CFAR threshold and by nulling out the region around the previously detected targets to avoid overlap of detections. The target of interest is then identified from amongst the N targets at each time instance using the target tracking algorithm described in §III-F.

Zero radial velocity: As mentioned before, to detect a target with zero radial velocity, the Doppler power corresponding to the zero frequency bin $\mathbf{P}_0$ needs to be analyzed. This is because, at low/zero radial velocity, the received power contributed by the target is shifted to the zero Doppler velocity bin. Let us consider an example. In Fig. 10, the Doppler power corresponding to one spatial bin, $\mathbf{T}$ is shown. Specifically, the time series $P_0[\mathbf{T}, s]$ and $P[\mathbf{T}, s]$ are shown with respect to s. It can be observed that when the target reaches a low radial velocity at the time indices highlighted in the figure, there is a shift in the Doppler power from $\mathbf{P}$ to $\mathbf{P}_0$. However, the application of CFAR target detection on $\mathbf{P}_0$ is not effective due to reflections from irrelevant static objects in the environment, usually referred to as clutter in the radar literature [32]. The noise level or the clutter can be estimated from the Doppler power of a particular spatial bin when the target is not present. Therefore, instead of estimating the noise from the adjacent spatial locations as in the CFAR technique, we estimate it from the median power of a particular bin over time. Let C_0^{th} denote the threshold for target detection on $\mathbf{P}_0$. It can be written as, $C_0^{\text{th}}[\mathbf{T}] = median(P_0[\mathbf{T}, .])$. The Doppler power matrix after CFAR threshold, $\mathbf{P}_0^c$ is written as:

$$P_0^c[\mathbf{T}, s] = \begin{cases} P_0[\mathbf{T}, s], & \text{if } P_0[\mathbf{T}, s] > C_0^{\text{th}}[\mathbf{T}] \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

Based on the above discussions, the target detection module is designed as follows. For each time instance, we identify a maximum of N non-overlapping probable locations of the target corresponding to the decreasing Doppler power from $\mathbf{P}_0^c$ and $\mathbf{P}^c$. The location coordinates corresponding to the spatial bins of the targets are stored in $\mathbf{T}_0^\star$ and $\mathbf{T}^\star$ respectively as described in Algorithm 1.

Algorithm 1 Target detection algorithm for *mmWrite*

for each time instance s **do**,
  for $i = 1 : N$ **do** ▷ N is the number of targets detected
    $\mathbf{T}_{\max} = \text{argmax}(P^c[\mathbf{T}, s])$
    if $P^c[\mathbf{T}_{\max}^{\mathbf{T}}, s] > 0$ **then**
      $\mathbf{T}^\star[i, s] = \mathbf{T}_{\max}$
    end if
    $P^c[\mathbf{T}_{\max} - \Delta : \mathbf{T}_{\max} + \Delta, s] \rightarrow Null$
  end for
  Repeat Step 2 to 8 with $\mathbf{P}_0^c$ to determine $\mathbf{T}_0^\star$
end for

E. Target Localization

From the previous stage, the location bin of the target can be determined from $\mathbf{T}^\star[., s]$ at every time step. Since the accuracy

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3066507, IEEE Internet of Things Journal

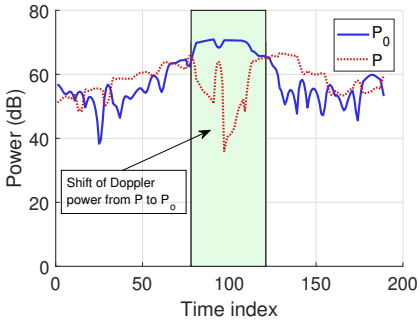
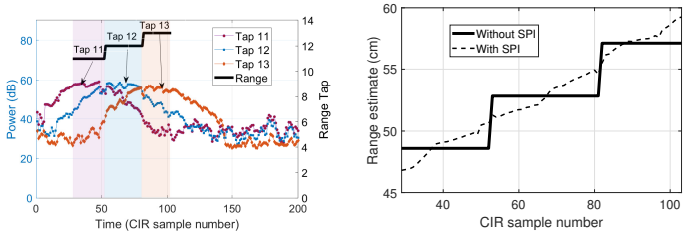
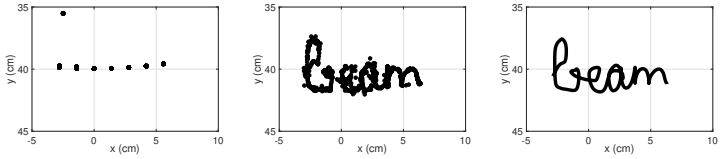


Fig. 10: Demonstration of the Doppler power shift from non-zero frequency bin to the zero frequency bin when the target has low or zero radial velocity. Observe that when the target reaches a low radial velocity at time index of about 75, the Doppler power shifts from the non-zero frequency bin to the zero frequency bin and then switches back when the target gains some radial velocity at a time index of about 120.



(a) (b)

Fig. 11: (a) Doppler power for different range taps with time for a straight line drawn along the range away from the device, (b) Range estimates before and after SPI.



(a) (b) (c)

Fig. 12: Illustration of the target localization and trajectory construction for the word "beam". (a) Discrete points of the trajectory from target detection, (b) Finer location estimates obtained from SPI and (c) Trajectory after DCT-based smoothing.

of the estimates is limited to 2 degrees in the azimuth and elevation dimensions and 4.25 cm in the range dimension from the design of the radar, a discontinuous/quantized trajectory is obtained for characters spanning only a few centimeters as shown in Fig. 12a. However, due to the smooth and continuous nature of handwriting, it has been observed that the Doppler power "flows" gradually from one bin to another. This fact is leveraged to obtain finer location estimates. The observation can be demonstrated by a simple experiment, as explained below.

A straight line is drawn away from the device, keeping the azimuth and elevation nearly zero, and the corresponding Doppler power in the range taps is analyzed. As shown in Fig. 11a, as the target moves away from the radar, the Doppler power shifts gradually from tap 11 (46.46 cm) to tap 13 (55.38 cm). According to our proposed approach, the tap corresponding to the maximum Doppler power is considered as the range of the target. Therefore, we get the discrete range estimates as shown in black. This observation motivated us to combine the information in adjacent taps to obtain an improved location estimate. However, using adjacent four values near the peak is not very useful. For example, during the period when the power corresponding to tap 13 is the maximum (orange line), the power corresponding to tap 11 (purple line) is already close to the noise level. This observation motivated us to use the SPI technique discussed in §II to improve the location estimates. In this work, SPI is applied to the range, azimuth, and elevation dimension independently. Fig. 11b shows the range estimates before and after SPI. Interpolation using the combined information from the three dimensions could improve the estimation, and such an analysis is left for future work.

The SPI operation has a significant effect on the performance of handwriting tracking as demonstrated in Fig. 12 since it largely boosts the spatial resolution. The trajectory of the word "beam" in Fig. 12a is formed by many overlapping points because of the discrete location taps (range, azimuth and elevation) while the trajectory formed after SPI in Fig. 12b is readable by the naked eye. Let $\mathbf{T}^{\star}_{a,b,c}[i,s] = [az^{\star}[i,s] + a, el^{\star}[i,s]+b, r^{\star}[i,s]+c]$ where a, b, c are the index offsets in the corresponding dimensions. The final estimates of azimuth, elevation and range coordinates are obtained from the SPI as:

$$az^{\star}_{\mathrm{SI}}[i,s] = \mathrm{SPI}\{P[\mathbf{T}^{\star}_{-1,0,0}[i,s],s], P[\mathbf{T}^{\star}_{0,0,0}[i,s],s], P[\mathbf{T}^{\star}_{1,0,0}[i,s],s]\}. \quad (14)$$

$$el^{\star}_{\mathrm{SI}}[i,s] = \mathrm{SPI}\{P[\mathbf{T}^{\star}_{0,-1,0}[i,s],s], P[\mathbf{T}^{\star}_{0,0,0}[i,s],s], P[\mathbf{T}^{\star}_{0,1,0}[i,s],s]\}. \quad (15)$$

$$r^{\star}_{\mathrm{SI}}[i,s] = \mathrm{SPI}\{P[\mathbf{T}^{\star}_{0,0,-1}[i,s],s], P[\mathbf{T}^{\star}_{0,0,0}[i,s],s], P[\mathbf{T}^{\star}_{0,0,1}[i,s],s]\}. \quad (16)$$

At time instant s, the location of the i^{th} moving target is given by $\mathbf{L}[i,s] = [az^{\star}_{\mathrm{SI}}[i,s], el^{\star}_{\mathrm{SI}}[i,s], r^{\star}_{\mathrm{SI}}[i,s]]$. Let the location estimates obtained from $\mathbf{T}^{\star}_{0}[i,s]$ be denoted by $\mathbf{L}_0[i,s]$. The ultimate trajectory is constructed using $\mathbf{L}$ and $\mathbf{L}_0$ in the next section.

F. Trajectory Construction

In the previous module, multiple target locations (N) have been detected at each time instance, s and stored in $\mathbf{L}$ and $\mathbf{L}_0$. In this module, a trajectory tracking algorithm is designed to extract the raw trajectory of the target of interest from $\mathbf{L}$ and $\mathbf{L}_0$. Following are some observations that will be used to build the algorithm.

1) Assuming the handwriting is on a flat surface and is smooth and continuous, the elevation angle of the writing object does not change abruptly. This fact is used to define continuity of target location at two successive time instances.
2) At any given time instance, the elevation angle corresponding to the target of interest, i.e., the tip/lower part of the writing tool, is the least among all the other detected target locations.
3) If a particular location is being continuously detected for many time instances in $\mathbf{L}$ and suddenly disappears, there is a possibility that the target has reached a low radial velocity and is detected in $\mathbf{L}_0$.

The target tracking algorithm is described in Algorithm 2. The target locations are read from $\mathbf{L}$ over time. The continuous trajectories are tracked and stored in a list. If a trajectory segment is continuous for more than a predefined number of time instances, then it is considered as a potential trajectory of the target of interest. If there are multiple potential trajectories, the lowest elevation trajectory is considered as the *mmWrite* trajectory. Additionally, if a trajectory suddenly becomes discontinuous after being detected as the *mmWrite* trajectory for a while, then we also check if the target location is detected in $\mathbf{L}_0$ to determine continuity. Some of the terms used are summarized below:

- CS (Continuity Score): Each detected target location is associated with a score abbreviated as CS. It is a scalar that indicates the length of the continuous trajectory detected in the previous instances of which the current location is a part.
- CS_{th}: Threshold on the continuity score
- CTL (Continuous Target List): List of target locations detected whose CS is greater than 0.
- Cont(a,b): Function to check if two locations a and b are continuous. Recall that two locations detected at successive time instances are said to be continuous if the elevation angle does not change significantly.

The trajectory tracking algorithm also removes outliers that result from sudden changes in the point of reflection or missed targets. The tracked target locations with time are stored in the vector $\mathbf{W}$. A raw trajectory can be formed by combining all the estimates. Such a trajectory is noisy, and an example is shown in Fig. 12b. To obtain a smoother trajectory, we propose a smoothing technique based on the Discrete Cosine Transform (DCT). The idea is to compute the DCT of the noisy data and inverse transform to the data domain while preserving only the significant coefficients. By doing so, we discard the insignificant coefficients that account for the noisy and irrelevant part of the signal [33], thereby obtaining a smooth trajectory. Mathematically, the recovered writing trajectory, $\hat{\mathbf{W}}$ is given by,

$$\hat{\mathbf{W}} = \text{IDCT}(\tau \text{DCT}(\mathbf{W})). \tag{17}$$

Here IDCT is the inverse discrete cosine transform, and τ is a diagonal matrix whose entries are given by,

$$\tau_{i,j=} \begin{cases} [1 + s(2 - 2\cos(\frac{(i-1)\pi}{n}))^2]^{-1} & i = j, \\ 0 & i \neq j, \end{cases}$$

Algorithm 2 Target tracking algorithm for *mmWrite*

for each time instance s **do**,
  W[s] $\leftarrow$ lowest elevation target in CTL with $CS > CS_{\text{th}}$ ▷ $\mathbf{W}$ stores the final target locations.
  for $j = 1 :$ length(CTL) **do** ▷ Update current entries of CTL.
    if any(cont($CTL(j)$, $L[:, s]$))==1 **then**
      $CTL(j).CS++$
    else
      if cont($CTL(j), W(s)$) **then** ▷ Target missed from $\mathbf{L}$.
        if any(cont($CTL(j)$,$L_0[:, s]$))==1 **then** ▷ Search for missed target location in $\mathbf{L}_0$.
          $CTL(j).CS++$
        else
          $CTL(j) \rightarrow []$ ▷ Target missed from $\mathbf{L}$ and $\mathbf{L}_0 \implies$ remove from CTL.
        end if
      end if
    end if
  end for
  for k=1:N **do** ▷ Add new entries to CTL.
    if any(cont($L[k, s], CTL(:)$))!=1 **then** ▷ New location discontinuous with CTL $\implies$ add to CTL.
      $loc.L = L[k, s]; loc.CS = 1;$
      $CTL(end + 1) = loc$
    end if
  end for

where n is the number of data points and s is the smoothing factor. The DCT coefficients of the data points are scaled by the matrix τ in the transform domain. The scaled coefficients are converted back to the data domain using the IDCT. The trajectory smoothing is achieved by the scaling matrix τ with the smoothing factor s and the cosine factor which imposes decreasing weights to the DCT coefficients. An example of the smoothed trajectory after DCT-based smoothing ($\hat{\mathbf{W}}$) is shown in Fig. 12c.

IV. Experimental evaluation

The performance of the proposed *mmWrite* system is evaluated in this section. Moreover, the effect of different parameters and experimental settings is discussed in detail. All the experiments are performed in a busy office environment using a similar setup as shown in Fig. 13, in which the writing surface and the radar lie on a flat surface. In the default setup, the height of the writing surface is adjusted to be around the zero elevation angle of the radar. Handwriting is performed so that the writing tool faces the radar and is the closest moving target to the radar. The writing tool is held at its upper portion to facilitate differentiation of the reflections from the writing tool and the hand/fingers. The handwriting itself is performed with natural writing speed using a marker on a paper/board. The CIR time series is recorded during the writing and is processed later using *mmWrite*. The recovered trajectories are subjected to different types of evaluations, which are described in detail below.

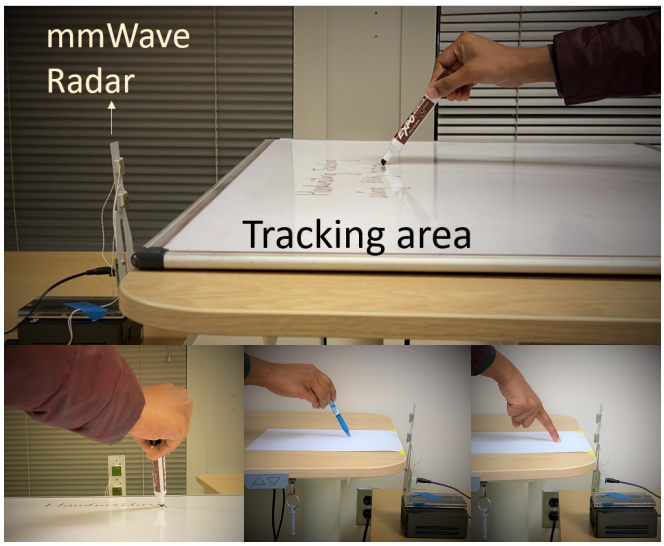


Fig. 13: Experimental setup for *mmWrite*. (Top) Writing on a whiteboard using a marker, (Bottom left to right) Front view of the setup showing the writing surface at nearly zero elevation, writing using a pen and hand.

A. Overall Performance

The performance of *mmWrite* is evaluated using three different approaches. Unless otherwise mentioned, the radar is placed at the origin $(0, 0)$.

Visual inspection and Shape matching: A valid handwriting tracking system should be able to preserve the relative shape of the handwriting trajectory that is legible to humans or recognizable by any standard handwriting recognition software. We assess this by visually comparing the ground-truth handwritten trajectories and the recovered trajectories by *mmWrite*. In this experiment, randomly selected English alphabets from the EMINST dataset [34] are printed and traced within an area about 5 cm $\times$ 5 cm on an A4 paper with a marker. Fig. 14 shows two versions of different English alphabets with the first, third and the fifth row showing the ground truth character images and the second, fourth and the sixth rows showing the recovered characters. It can be confirmed that the relative shapes of the alphabets are well preserved by *mmWrite*, which are visually similar to the ground-truth handwriting and can be recognized by humans confidently. Additionally, we also evaluated the shape similarity using shape context as discussed in [35]. Classification using shape context achieved an accuracy of 96.6% on the characters shown in Fig. 14.

Character recognition accuracy: Given the encouraging results as in Fig. 14, it would be interesting to see how good the recovered characters could be recognized by standard handwriting recognition tools. To evaluate that, we wrote English alphabets in different sized grids and at different distances from the device. For each scenario, we randomly picked 50 alphabets from the EMINST dataset and traced their trajectories on a paper. The trajectories obtained by *mmWrite* are then exported from MATLAB to control the mouse pointer and are then fed to a standard handwriting recognition software (myScript). The character recognition accuracy reported by the software is recorded and shown in Fig. 15. It can be observed from the figure that the accuracy decreases with distance from the device and increases with the scale of handwriting, as expected. This is because of the decreasing cross-range accuracy. For instance, the character recognition accuracy for characters written within 3 cm x 3 cm is 80% and 72% at distances 20 cm and 30 cm respectively. The accuracy for the same distance of 20 cm is 80% and 82% for scales of 3 cm x 3cm and 5 cm x 5 cm respectively. As will be shown later in Fig. 21, the tracked characters, even written in really small scales (1 cm $\times$ 1 cm), can still be easily read by human eyes, although they might not be accurately recognized by standard software.

Tracking accuracy: Now we quantitatively evaluate the tracking accuracy of *mmWrite*. To do so, we trace a circle and a triangle (both predefined with known size and shape which serve as the ground truth trajectory) multiple times, and reconstruct the trajectories. Due to lack of time synchronization between the handwritten trajectories and the *mmWrite* estimates, we report the median of the minimum projection errors between the two versions of trajectories as the tracking error. Two instances of the ground truth and the projected trajectories are shown in Fig. 16a and Fig. 16b.

It is known that the distance from the device affects the tracking performance due to the reduced cross-range accuracy. This can be observed from the CDF of the tracking errors shown in Fig. 16c where the tracking error increases with distance from the radar (R). To further gauge the influence of range and azimuth angles on the tracking error, we repeat the above experiment for different ranges and azimuth angles. Fig. 17 shows the resultant error map. The median tracking error increases from about 3 mm at a range of 20 cm to 40 mm at a range of 3 m. The tracking error also slightly increases with the azimuth angle at any given range. The reflected power from the tiny writing object is too weak to be detected by the radar, beyond a range of 3 m. Also, the target cannot be detected beyond an azimuth angle of 50-55 degrees due to the hardware limitations. Nevertheless, Fig. 17 has demonstrated that the *mmWrite* system can already cover an area of about 8 m^2, about 10$\times$ larger than the state-of-the-art system using mmWave radio [36].

B. Micro-benchmarks

Handwriting tracking on different surfaces: *mmWrite* can be used to track handwriting on flat surfaces of different materials. This is because only the signals reflected off the writing object are analyzed by *mmWrite*, and the material of the writing surface is irrelevant. *mmWrite* can, therefore, transform any flat surface to a potential writing surface. Fig. 18 shows some instances of handwriting tracked on different materials such as paper, glass, and cardboard.

Handwriting tracking with different writing objects: Intuitively, we expect the writing object's properties to have a more direct effect on the tracking performance than the writing surface. To analyze this, we have performed tracking with different writing objects: a marker, a pen, a pencil, and a

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3066507, IEEE Internet of Things Journal

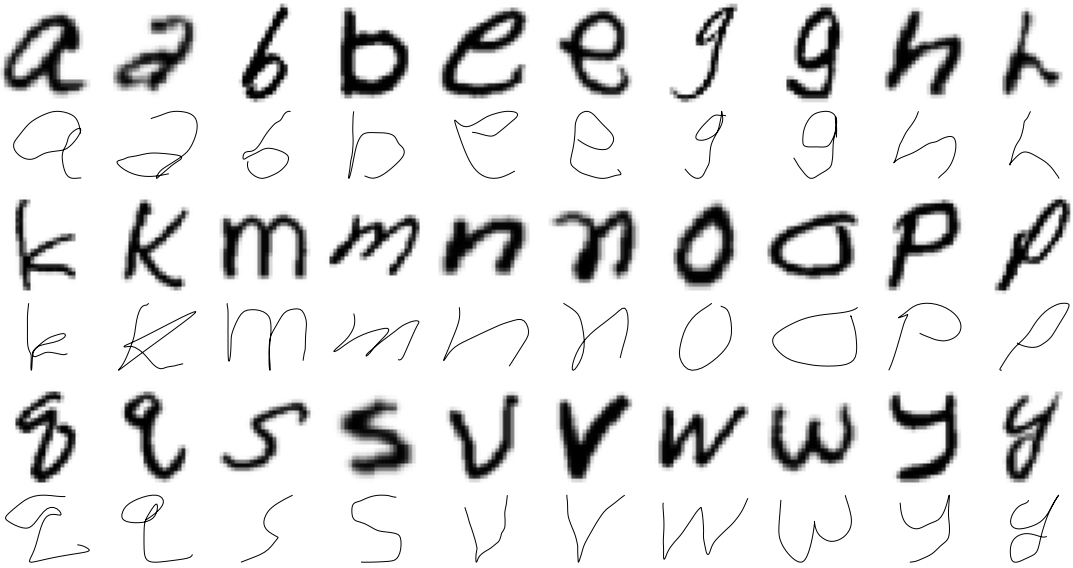


Fig. 14: Visual comparison of ground truth characters and recovered characters. The characters in bold are picked from EMINST dataset and serve as the ground truth (row 1, 3, and 5). The corresponding recovered characters are shown in row 2, 4, and 6.

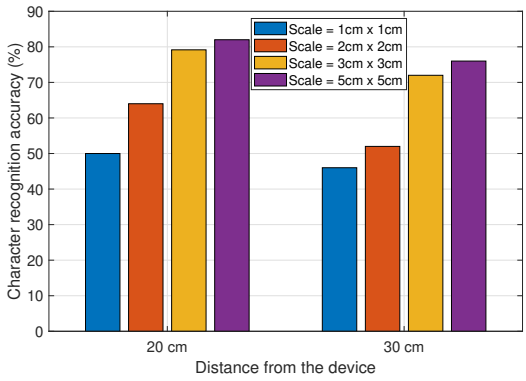


Fig. 15: Character recognition accuracy at different distances from the device with characters written in different scales.

human finger. We have traced a circle with these objects, and the raw trajectory points, **W** are shown in Fig. 20. It can be observed that a smooth and continuous trajectory is obtained in the case of a marker, while the trajectory points obtained in the case of a hand/finger gave noisy estimates. This behavior can be associated with different incidence points on the target and the non-rigid nature of the hand. The great performance with a marker will be particularly useful by transforming traditional whiteboards into interactive ones using *mmWrite*.

Handwriting recognition with different users: To study the user dependency on the performance of *mmWrite*, we have requested 5 different users to draw 50 characters at a range of about 30 cm and in an area spanning about 7 cm x 7 cm. The corresponding recognition accuracy is reported as shown in Fig. 19. It can be observed that the average character recognition is about 91% over different users.

Scale of handwriting: Apparently, the performance of *mmWrite* depends on the scale of handwriting. The higher the scale, the easier it is to identify the relative shape of the trajectory. In Fig. 15, the character recognition accuracy for characters written in different scales is shown. To explore the limit of the scale of handwriting that can be retrieved by *mmWrite*, experiments were performed with characters spanning a few millimeters at a distance of 20 cm from the radar. Fig. 21a shows the ground truth trajectory of the word "scale" with the grid size of 1 cm x 1 cm and Fig. 21b shows the obtained trajectory using *mmWrite*. It has been observed that characters written in a grid of size as small as 1 cm x 1 cm can be recovered and the resultant trajectories can be easily recognized by humans.

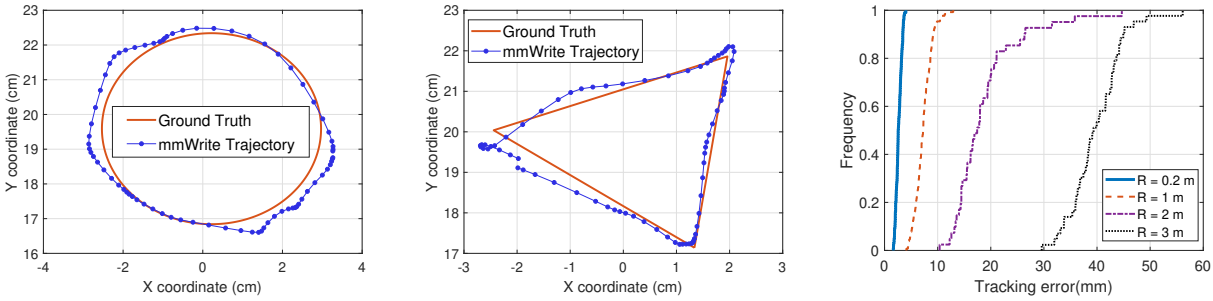
Repeatability of the trajectories: The absolute localization of the target enables *mmWrite* to generate precise estimates of the handwriting trajectories. To demonstrate the repeatability of *mmWrite* trajectories, the word "repeat" is traced several times on a paper and the trajectories are reconstructed. Fig. 22a shows the picture of the traced word on a paper and Fig. 22b shows the corresponding *mmWrite* trajectories. It can be observed that the trajectories overlap on each other, demonstrating the consistent precision of the estimated trajectories.

Number of points on the trajectory: *mmWrite* reconstructs the handwriting trajectory by estimating the location of the moving/writing target at different time instances and by connecting all the location estimates. For a given length of the trajectory, if the target localization is performed at shorter intervals, a greater number of points are obtained, capturing more abrupt changes in the trajectory. However, this performance enhancement is achieved at the cost of higher computational complexity.

The number of points on the recovered trajectory is dependent on two factors:

- **The speed of handwriting**: With a fixed sampling frequency of the CIR time series, faster handwriting results

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3066507, IEEE Internet of Things Journal



(a) (b) (c)

Fig. 16: Ground truth and the reconstructed trajectories of a (a) circle and (b) triangle.(c) CDF of tracking error.

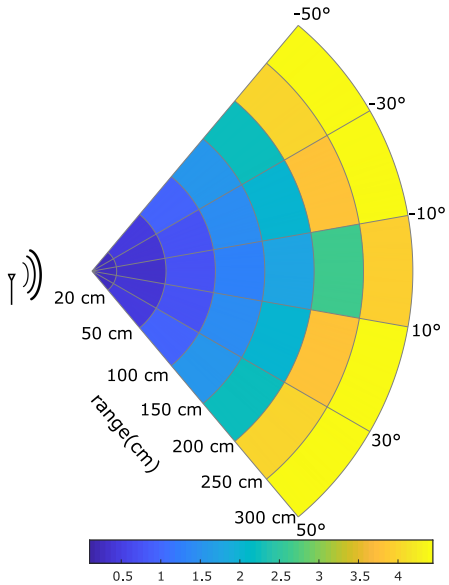


Fig. 17: Tracking error at different range and azimuth angles from the radar.

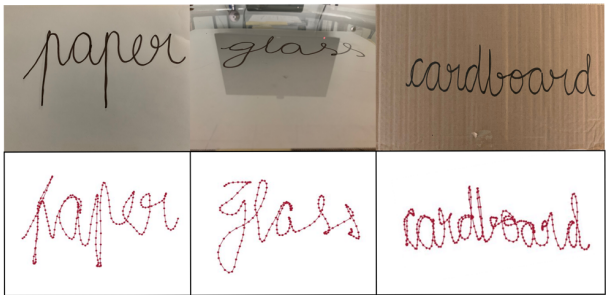


Fig. 18: Handwriting tracking on different materials

in fewer points on the trajectory compared to slower handwriting. Fig. 23 shows the recovered trajectories for letter 'm' written with two extreme speeds. Fig. 23a shows handwriting with very slow speed while Fig. 23b shows the trajectory corresponding to fast handwriting. Observe that for the faster speed, the number of points on the trajectory is less and the trajectory becomes smoother. However, faster speed may not capture the minute details

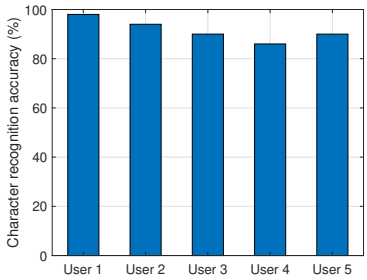


Fig. 19: Character recognition accuracy for different users.

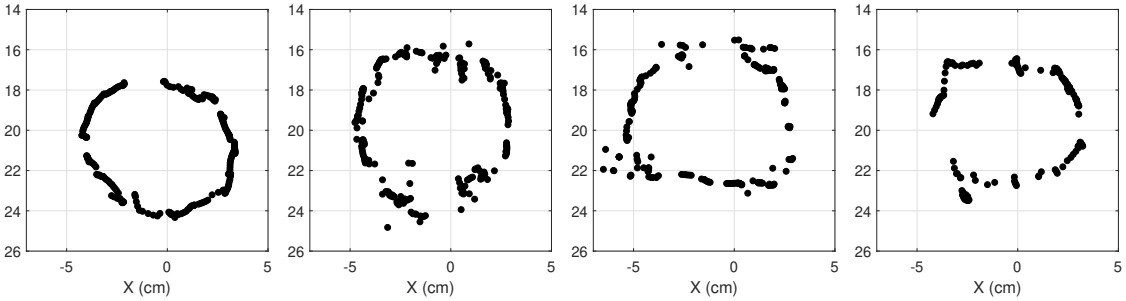
of the trajectories.

- **Step size in the windowed Fourier transform** (w_s): From the beamformed CIR time series, *mmWrite* computes the STFT along the slow-time dimension. As the step size of the moving window (w_s) decreases, target localization is performed more frequently, resulting in a larger number of points on the trajectory. For a step size of w_s, the sampling frequency of the points on the trajectory is $1/(w_s \text{ x } T_b)$, where T_b is the duration of the burst as discussed in §II-A. For instance, if the step size, $w_s = 20$ and $T_b = 400\ \mu s$, the sampling frequency is 125 Hz. For a fixed $T_b = 400\ \mu s$ the trajectories of writing letter 'n' with different w_s are shown in Fig. 24. As w_s increases, the sampling frequency decreases and lesser number of points are obtained on the trajectory. This causes smoothening of minute details in the trajectory.

Degree of smoothness: The degree of smoothness achieved by the DCT-based smoothing depends on the smoothing parameter s. For a given number of points on the trajectory, the higher the value of s, the smoother the trajectory. However, if the value of s is too large, necessary details in the relative shape of the trajectory can be smoothed out, which is undesired. For instance, in Fig. 25, the smoothed trajectories with different values of the smoothing parameter s are shown. In this example, for $s = 1500$, the letter "h" is barely recognizable. *mmWrite* defaults with $s = 2$ in this work.

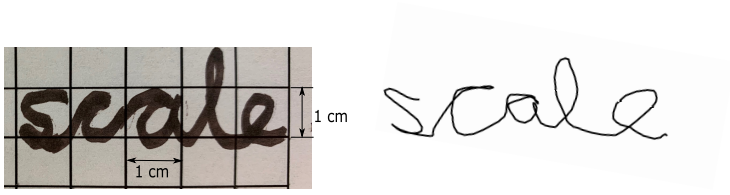
Complexity and Latency: Let N be the total number of points on a trajectory. The complexity for each of the modules in *mmWrite* is shown in Table. I. Most modules include multiple linear-complexity operations and have an overall complexity of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3066507, IEEE Internet of Things Journal



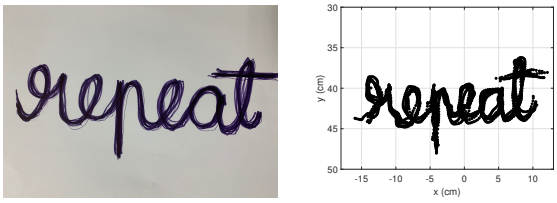
(a) Marker (b) Pen (c) Pencil (d) Hand

Fig. 20: Illustration of the trajectory points detected for different writing objects. (a) Marker, (b) Pen, (c) Pencil and (d) Finger.



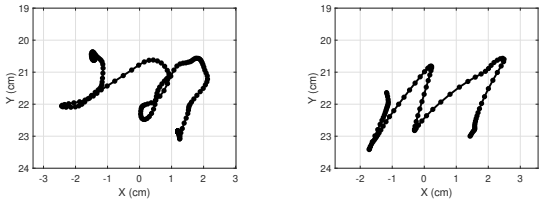
(a) (b)

Fig. 21: Demonstration of handwriting tracking for characters within 1 cm. (a) Actual trajectory in a grid of 1 cm x 1 cm, (b) *mmWrite* trajectory.



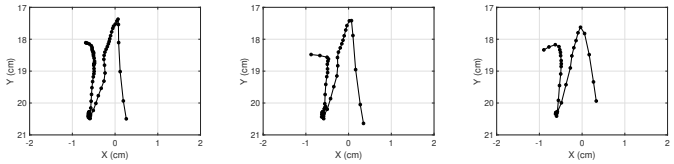
(a) (b)

Fig. 22: Demonstration of *mmWrite* precision. (a) Ground truth of word "repeat", (b) *mmWrite* result.



(a) (b)

Fig. 23: Recovered handwriting trajectories with extreme speeds of handwriting. (a) Very slow and (b) Very fast handwriting.



(a) (b) (c)

Fig. 24: Recovered handwriting trajectories with different w_s. (a) w_s = 20 (b) w_s = 30 (c) w_s = 50.

$O(N)$. Although the Doppler velocity extraction module uses the FFT operation, which has a complexity of $O(NlogN)$, the transformation is performed along the *slow-time* dimension with a duration of w_l resulting in $O(1)$ operations for each point. The trajectory construction module uses the DCT operation, which has a complexity of $O(NlogN)$. Overall, *mmWrite* has a complexity of $O(NlogN)$. This can be further optimized by smoothening discontinuous strokes separately, which is part of future work.

TABLE I: Complexity for different modules in *mmWrite*.

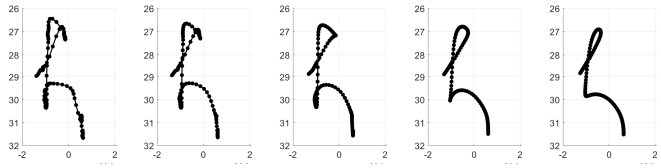
Module	Complexity
Background Subtraction	$O(N)$
Digital Beamforming	$O(N)$
Doppler Velocity Extraction	$O(N)$
Target Detection	$O(N)$
Target Localization	$O(N)$
Trajectory Construction	$O(NlogN)$

The current version of *mmWrite* uses offline processing and records the complete CIR time series before producing the handwriting trajectory. If the static background is recorded beforehand, the estimated time in MATLAB to generate one raw trajectory point is about 0.4 seconds on a Windows machine with an Intel Core i7 processor. We can obtain a substantial reduction in the computation time by using parallel processing and GPUs.

V. Related Works

Handwriting tracking Handwriting tracking can be achieved in a device-based [5], [6] or device-free manner [7], [36], [37]. Considering user-convenience, device-free or passive approaches have gained more attention over the years. Passive handwriting tracking has been attempted by researches in the past using different approaches such as acoustic, visual, WiFi, RFID, and millimeter waves. A summary of the past works using different modalities is presented in Table. II. Acoustic signal-based approaches use the inbuilt speaker and microphone of the smartphones to derive the sound signal reflected by the moving object. Although these approaches could track handwriting at millimeter-scale accuracy, the performance deteriorates severely with distance, and the oper-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3066507, IEEE Internet of Things Journal



(a) s=0.5 (b) s=5 (c) s=50 (d) s=500 (e) s=1500

Fig. 25: Smoothed trajectories for different values of the smoothing parameter s.

ational range is usually within 50 cm from the device [9], [38], [39]. Vision-based methods use a camera to identify and track the fingertip/arm to reconstruct the handwriting trajectory [7], [40]. Intuitively, the tracking accuracy of the vision-based approaches depends on the pixel resolution and the distance from the camera. These approaches are limited by privacy interests and the requirement for ambient light. On the other hand, WiFi-based approaches estimate the displacement using Fresnel zone model [41] or by tracking the phase of the CSI [37]. The transceiver geometry is then exploited to obtain the 2D location of the target. WiDraw [42] uses the Angle of arrival (AoA) of the line-of-sight signals from the densely deployed transceivers to track the motion of the hand in the air. Although WiFi based approaches appear attractive in terms of coverage, the tracking accuracy is of the order of centimeters or decimeters, which is not adequate for fine-grained handwriting tracking [37], [42]. Compared to the above modalities, millimeter wave-based methods seem to be an attractive option both in terms of coverage and tracking accuracy [36] [43].

Among the millimeter wave based approaches, *mmWrite* is the first work to reuse a commodity millimeter wave radio as a radar for handwriting tracking. The most relevant past work is *mTrack*, which uses a custom-built 60 GHz software-radio platform (consisting three transceivers) and tracks the writing object by using the phase of the received signal. While *mTrack* relies on phase changes and triangulation, we leverage more radar capabilities in target localization and Doppler speed estimation, as compared in Table. III. It can be observed that *mmWrite* can achieve lesser tracking error and larger coverage than *mTrack*, while requiring minimal infrastructure support.

Handwriting recognition The field of handwriting tracking is closely linked to that of handwriting recognition. While the end goal of handwriting tracking is to reconstruct the trajectory, handwriting recognition further intends to predict/identify the character. Handwriting recognition can be achieved in two different approaches. The first set of approaches recover the handwriting trajectory as an intermediate step and identify the characters using handwriting recognition modules/classifiers [42], [45]. The second set of approaches extract features representative of the characters and use learning-based techniques to classify them. These approaches suffer from the pattern inconsistency problem, i.e., the extracted features are location and environment-dependent, limiting the practical usage and the allowed set of characters [10], [46]. The current research, therefore, is focused on recovering the handwriting trajectory itself allowing unrestricted device placement and the set of trajectories.

Gesture recognition and passive motion tracking Wireless signals like WiFi and mmWave have also been widely explored for gesture recognition and passive motion tracking. These works either set up multiple transceivers with a specific geometric arrangement for precise gesture tracking [37], [41], or do not need special setup but only allow gesture recognition from a predefined set of gestures [47]–[49]. The latest research resorts to deep learning to improve cross-domain recognition accuracy, but does not fundamentally improve motion tracking resolution [50], [51]. These works, however, could not achieve the required millimeter accuracy for handwriting tracking due to the intense multipath that corrupts the signal reflected from the hand.

Indoor passive motion tracking is also achieved with 5 GHz WiFi. For this, parameters such as Angle-of-Arrival, Doppler velocity and Time-of-flight [52]–[54] are estimated from the CSI measurements. Millimeter wave has been used to track one or more people/robots by exploiting the target tracking capabilities of the radar and the phased antenna array structure [55], [56]. However, these works aim at tracking the human/robot body as a whole and the tracking accuracy is not sufficient for the handwriting application.

VI. Discussions and Future Work

In this work, we built a prototype of a handwriting tracking system. At this stage, *mmWrite* can be further improved in the following ways.

- We assumed a single moving target in this work. The algorithm can be extended to multiple moving targets by repeating the target detection, localization, and tracking steps at each time instance for the desired number of targets. Spatial resolution plays a crucial role in the performance of multiple target tracking. The tracking error could be further reduced using multiple radios, for instance, by using a triangulation method.
- We do not deal with multiple/discontinuous strokes in this work. With the current version of *mmWrite*, the recovered trajectories are projected onto a 2D plane. Therefore, discontinuous writing strokes would still appear continuous. One way to handle multiple strokes is to exploit the elevation information of the target, i.e., by discarding the trajectory points above a pre-defined threshold. Such a module is beyond the scope of this manuscript and is left for future work.

VII. Conclusion

In this work, we proposed *mmWrite*, a handwriting tracking system using a single commercial 60 GHz radar. A complete pipeline of the handwriting tracking system from detecting the writing object to the post-processing steps is presented. It has been proved that the relative shape of the handwritten characters/words is preserved, which can be read by a human or fed to a handwriting recognition software. The performance of *mmWrite* is evaluated and discussed under different experimental conditions. With *mmWrite*, we envision smart environments in which any flat region can be a writing

TABLE II: Summary of passive handwriting tracking systems

Modality	Reference	Method	Hardware	Coverage	Tracking Accuracy
Acoustic	Strata [38]	Phase tracking	Smart phone	40 cm	3 mm
	LLAP [9]	Phase tracking of sound signals	Smart phone	30 cm	4.6 mm
	FingerIO [39]	Echo profile and OFDM	Smart phone	0.25 m^2	8 mm
Vision	[7]	HMM and multi camera 3D tracking	Camera/Kinect	N/A	N/A
	[40]	Fingertip detection and tracking	USB PC Camera	50 cm*	N/A
WiFi	FingerDraw [41]	Fresnel zone, triangulation	3 transceivers	100 cm	1.27 cm
	AirDraw [37]	Phase tracking and triangulation	3 transceivers	1.5 m	2.2 cm
	WiDraw [42]	Analysis of RSSI with direction	30 transceivers	2 feet	5 cm
RFID	RF-Finger [44]	Reflection features from tag array	RFID tags & readers	1 m*	~2 cm
mmWave	mTrack [36]	Phase tracking and triangulation	3 transceivers	1 m^2	7 mm
	mmWrite	**Target detection & localization**	**One 60 GHz radar**	**8** m^2	**2.8 mm**

$\star$ Approximate values based on the experimental setup.

TABLE III: Comparison between *mTrack* and *mmWrite*

Reference	Hardware	Method	Absorbers	Calibration	Coverage	Error at 20 cm
mTrack [36]	1 Tx, 2 Directional horn antenna	Phase tracking & triangulation	Required	Required	1 m^2	7 mm
mmWrite	Phased antenna array	Target localization	Not required	Not required	~8 m^2	2.8 mm

surface, and the space for HCI can be extended beyond the touch screens.

REFERENCES

[1] J. Lien, N. Gillian, M. E. Karagozler, P. Amihood, C. Schwesig, E. Olson, H. Raja, and I. Poupyrev, "Soli: Ubiquitous Gesture Sensing with Millimeter Wave Radar," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, pp. 1–19, 2016.

[2] B. Fang, C. H. Leung, Y. Y. Tang, K. Tse, P. C. Kwok, and Y. Wong, "Off-line signature verification by the tracking of feature and stroke positions," *Pattern recognition*, vol. 36, no. 1, pp. 91–101, 2003.

[3] N. Arica and F. T. Yarman-Vural, "An Overview of Character Recognition Focused on Off-Line Handwriting," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 31, no. 2, pp. 216–233, 2001.

[4] C. C. Tappert, C. Y. Suen, and T. Wakahara, "The State of the Art in On-Line Handwriting Recognition," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 12, no. 8, pp. 787–808, 1990.

[5] S. Agrawal, I. Constandache, S. Gaonkar, R. Roy Choudhury, K. Caves, and F. DeRuyter, "Using Mobile Phones to Write in Air," in *Proceedings of the 9th international conference on Mobile systems, applications, and services*. ACM, 2011, pp. 15–28.

[6] J. Wang, D. Vasisht, and D. Katabi, "RF-IDraw: Virtual Touch Screen in the Air Using RF Signals," in *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 4. ACM, 2014, pp. 235–246.

[7] A. Schick, D. Morlock, C. Amma, T. Schultz, and R. Stiefelhagen, "Vision-Based Handwriting Recognition for Unrestricted Text Input in Mid-Air," in *Proceedings of the 14th ACM international conference on Multimodal interaction*, 2012, pp. 217–220.

[8] T. Murata and J. Shin, "Hand Gesture and Character Recognition Based on Kinect Sensor," *International Journal of Distributed Sensor Networks*, vol. 10, no. 7, p. 278460, 2014.

[9] W. Wang, A. X. Liu, and K. Sun, "Device-Free Gesture Tracking Using Acoustic Signals," in *Proceedings of the 22*6nd *Annual International Conference on Mobile Computing and Networking*. ACM, 2016, pp. 82–94.

[10] M. Chen, P. Yang, J. Xiong, M. Zhang, Y. Lee, C. Xiang, and C. Tian, "Your Table Can Be an Input Panel: Acoustic-based Device-Free Interaction Recognition," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 3, no. 1, p. 3, 2019.

[11] H. Du, P. Li, H. Zhou, W. Gong, G. Luo, and P. Yang, "WordRecorder: Accurate Acoustic-based Handwriting Recognition Using Deep Learning," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 2018, pp. 1448–1456.

[12] K. J. R. Liu and B. Wang, *Wireless AI: Wireless Sensing, Positioning, IoT, and Communications*. Cambridge University Press, 2019.

[13] B. Wang, Q. Xu, C. Chen, F. Zhang, and K. J. R. Liu, "The Promise of Radio Analytics: A future paradigm of wireless positioning, tracking, and sensing," *IEEE Signal Processing Magazine*, vol. 35, no. 3, pp. 59–80, 2018.

[14] J. Liu, G. Teng, and F. Hong, "Human Activity Sensing with Wireless Signals: A Survey," *Sensors*, vol. 20, no. 4, p. 1210, 2020.

[15] Y. Ma, G. Zhou, and S. Wang, "WiFi Sensing with Channel State Information: A Survey," *ACM Computing Surveys (CSUR)*, vol. 52, no. 3, pp. 1–36, 2019.

[16] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of Wireless Indoor Positioning Techniques and Systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 6, pp. 1067–1080, 2007.

[17] F. Zhang, C. Wu, B. Wang, H.-Q. Lai, Y. Han, and K. J. R. Liu, "WiDetect: Robust Motion Detection with a Statistical Electromagnetic Model," vol. 3, no. 3. ACM, 2019, p. 122.

[18] S. D. Regani, Q. Xu, B. Wang, M. Wu, and K. J. R. Liu, "Driver Authentication for Smart Car Using Wireless Sensing," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 2235–2246, 2019.

[19] J. Liu, Y. Wang, Y. Chen, J. Yang, X. Chen, and J. Cheng, "Tracking Vital Signs During Sleep Leveraging Off-the-shelf WiFi," in *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2015, pp. 267–276.

[20] C. Wu, F. Zhang, Y. Hu, and K. J. R. Liu, "GaitWay: Monitoring and Recognizing Gait Speed Through the Walls," *IEEE Transactions on Mobile Computing*, 2020.

[21] Y. Wang, J. Liu, Y. Chen, M. Gruteser, J. Yang, and H. Liu, "E-eyes: Device-free Location-oriented Activity Identification Using Fine-grained WiFi Signatures," in *Proceedings of the 20th annual international conference on Mobile computing and networking*, 2014, pp. 617–628.

[22] C. Wu, F. Zhang, B. Wang, and K. J. R. Liu, "mSense: Towards Mobile Material Sensing with a Single Millimeter-Wave Radio," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 4, no. 3, pp. 1–20, 2020.

[23] F. Wang, F. Zhang, C. Wu, B. Wang, and K. J. R. Liu, "ViMo: Multi-person Vital Sign Monitoring using Commodity Millimeter Wave Radio," *IEEE Internet of Things Journal*, 2020.

[24] T. Link Technologies Co. Ltd. (2020) Talon ad7200 multi-band wi-fi router. [Online]. Available: https://www.tp-link.com/us/home-networking/wifi-router/ad7200/

[25] Z. Meng, S. Fu, J. Yan, H. Liang, A. Zhou, S. Zhu, H. Ma, J. Liu, and N. Yang, "Gait Recognition for Co-Existing Multiple People Using Millimeter Wave Sensing," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, 2020, pp. 849–856.

[26] T. Wang, D. Zhang, Y. Zheng, T. Gu, X. Zhou, and B. Dorizzi, "C-FMCW Based Contactless Respiration Detection Using Acoustic Signal," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 4, pp. 1–20, 2018.

[27] K. Qian, C. Wu, F. Xiao, Y. Zheng, Y. Zhang, Z. Yang, and Y. Liu, "Acousticcardiogram: Monitoring Heartbeats Using Acoustic Signals on Smart Devices," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 2018, pp. 1574–1582.

[28] L. Zheng, M. Lops, Y. C. Eldar, and X. Wang, "Radar and Communication Coexistence: An Overview: A Review of Recent Methods," *IEEE Signal Processing Magazine*, vol. 36, no. 5, pp. 85–99, 2019.

[29] L. Svilainis and V. Dumbrava, "Analysis of the interpolation techniques for time-of-flight estimation," *Ultragarsas" Ultrasound"*, vol. 63, no. 4, pp. 25–29, 2008.

[30] X. Lai and H. Torp, "Interpolation Methods for Time-Delay Estimation Using Cross-Correlation Method for Blood Velocity Measurement," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 46, no. 2, pp. 277–290, 1999.

[31] B. D. Van Veen and K. M. Buckley, "Beamforming: A Versatile Approach to Spatial Filtering," *IEEE assp magazine*, vol. 5, no. 2, pp. 4–24, 1988.

[32] M. A. Richards, *Fundamentals of Radar Signal Processing*. Tata McGraw-Hill Education, 2005.

[33] D. Garcia, "Robust smoothing of gridded data in one and higher dimensions with missing values," *Computational statistics & data analysis*, vol. 54, no. 4, pp. 1167–1178, 2010.

[34] G. Cohen, S. Afshar, J. Tapson, and A. van Schaik, "EMNIST: Extending MNIST to handwritten letters," in *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2017, pp. 2921–2926.

[35] S. Belongie, J. Malik, and J. Puzicha, "Shape Matching and Object Recognition Using Shape Contexts," *IEEE transactions on pattern analysis and machine intelligence*, vol. 24, no. 4, pp. 509–522, 2002.

[36] T. Wei and X. Zhang, "mTrack: High-Precision Passive Tracking Using Millimeter Wave Radios," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. ACM, 2015, pp. 117–129.

[37] Z. Han, Z. Lu, X. Wen, J. Zhao, L. Guo, and Y. Liu, "In-Air Handwriting by Passive Gesture Tracking using Commodity WiFi," *IEEE Communications Letters*, 2020.

[38] S. Yun, Y.-C. Chen, H. Zheng, L. Qiu, and W. Mao, "Strata: Fine-Grained Acoustic-based Device-Free Tracking," in *Proceedings of the 15th annual international conference on mobile systems, applications, and services*, 2017, pp. 15–28.

[39] R. Nandakumar, V. Iyer, D. Tan, and S. Gollakota, "FingerIO: Using Active Sonar for Fine-Grained Finger Tracking," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 2016, pp. 1515–1525.

[40] L. Jin, D. Yang, L.-X. Zhen, and J.-C. Huang, "A Novel Vision-based Finger-Writing Character Recognition System," *Journal of Circuits, Systems, and Computers*, vol. 16, no. 03, pp. 421–436, 2007.

[41] D. Wu, R. Gao, Y. Zeng, J. Liu, L. Wang, T. Gu, and D. Zhang, "FingerDraw: Sub-wavelength Level Finger Motion Tracking with WiFi Signals," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 4, no. 1, pp. 1–27, 2020.

[42] L. Sun, S. Sen, D. Koutsonikolas, and K.-H. Kim, "WiDraw: Enabling Hands-free Drawing in the Air on Commodity WiFi Devices," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, 2015, pp. 77–89.

[43] S. D. Regani, C. Wu, F. Zhang, B. Wang, M. Wu, and K. J. R. Liu, "Handwriting Tracking Using 60 GHz mmWave Radar," in *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*. IEEE, 2020, pp. 1–6.

[44] C. Wang, J. Liu, Y. Chen, H. Liu, L. Xie, W. Wang, B. He, and S. Lu, "Multi-Touch in the Air: Device-Free Finger Tracking and Gesture Recognition via COTS RFID," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 2018, pp. 1691–1699.

[45] M. Liwicki, A. Graves, S. Fernàndez, H. Bunke, and J. Schmidhuber, "A Novel Approach to On-Line Handwriting Recognition Based on Bidirectional Long Short-Term Memory Networks," in *Proceedings of the 9th International Conference on Document Analysis and Recognition, ICDAR 2007*, 2007.

[46] Z. Guo, F. Xiao, B. Sheng, H. Fei, and S. Yu, "WiReader: Adaptive Air Handwriting Recognition Based on Commercial Wi-Fi Signal," *IEEE Internet of Things Journal*, 2020.

[47] H. Abdelnasser, M. Youssef, and K. A. Harras, "WiGest: A Ubiquitous WiFi-based Gesture Recognition System," in *2015 IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 2015, pp. 1472–1480.

[48] Z. Tian, J. Wang, X. Yang, and M. Zhou, "WiCatch: A Wi-Fi Based Hand Gesture Recognition System," *IEEE Access*, vol. 6, pp. 16 911–16 923, 2018.

[49] Q. Pu, S. Gupta, S. Gollakota, and S. Patel, "Whole-Home Gesture Recognition Using Wireless Signals," in *Proceedings of the 19th annual international conference on Mobile computing & networking*, 2013, pp. 27–38.

[50] J. Zhang, Z. Tang, M. Li, D. Fang, P. Nurmi, and Z. Wang, "CrossSense: Towards Cross-Site and Large-Scale WiFi Sensing," in *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, 2018, pp. 305–320.

[51] Y. Zheng, Y. Zhang, K. Qian, G. Zhang, Y. Liu, C. Wu, and Z. Yang, "Zero-Effort Cross-Domain Gesture Recognition with Wi-Fi," in *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*, 2019, pp. 313–325.

[52] X. Li, D. Zhang, Q. Lv, J. Xiong, S. Li, Y. Zhang, and H. Mei, "IndoTrack: Device-Free Indoor Human Tracking with Commodity Wi-Fi," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 3, pp. 1–22, 2017.

[53] K. Qian, C. Wu, Z. Yang, Y. Liu, and K. Jamieson, "Widar: Decimeter-Level Passive Tracking via Velocity Monitoring with Commodity Wi-Fi," in *Proceedings of the 18th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2017, pp. 1–10.

[54] K. Qian, C. Wu, Y. Zhang, G. Zhang, Z. Yang, and Y. Liu, "Widar2. 0: Passive Human Tracking with a Single Wi-Fi Link," in *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*, 2018, pp. 350–361.

[55] C. Wu, F. Zhang, B. Wang, and K. J. R. Liu, "mmTrack: Passive Multi-Person Localization Using Commodity Millimeter Wave Radio," in *IEEE INFOCOM*, 2020.

[56] D. Garcia, J. O. Lacruz, P. J. Mateo, and J. Widmer, "POLAR: Passive object localization with IEEE 802.11 ad using phased antenna arrays," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2020, pp. 1838–1847.