OPEN ACCESS

University of Dundee

University of Dundee

A high-speed multi-purpose software defined radar for near-field applications

Li, Wenda; Tang, Chong; Vishwakarma, Shelly; Shi, Fangzhan; Piechocki, Robert; Chetty, Kevin

Published in:
IEEE Radar Conference 2022

DOI:
10.1109/radarconf2248738.2022.9764260

Publication date:
2022

Document Version
Peer reviewed version

Link to publication in Discovery Research Portal

Citation for published version (APA):
Li, W., Tang, C., Vishwakarma, S., Shi, F., Piechocki, R., & Chetty, K. (2022). A high-speed multi-purpose software defined radar for near-field applications. In *IEEE Radar Conference 2022: Conference Proceedings* IEEE. https://doi.org/10.1109/radarconf2248738.2022.9764260

General rights
Copyright and moral rights for the publications made accessible in Discovery Research Portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 21. May. 2025

Li, Wenda et al. "A high-speed multi-purpose software defined radar for near-field applications". IEEE Radar Conference 2022: Conference Proceedings. New York City, NY: IEEE. 2022. https://doi.org/10.1109/radarconf2248738.2022.9764260

A High-Speed Multi-Purpose Software Defined Radar for Near-Field Applications

Wenda Li*, Chong Tang*, Shelly Vishwakarma*, Fangzhan Shi*, Robert Piechocki†, Kevin Chetty*
*Department of Security and Crime Science, University College London, UK
†Department of Electronic and Electrical Engineering, University of Bristol, UK
{wenda.li, chong.tang.18, s.vishwakarma, fangzhan.shi.17, k.chetty,}@ucl.ac.uk, r.j.piechocki@bristol.ac.uk

***Abstract*—Software Defined Radar (SDRadar) is a unique radar system, where most of its processing, like filtering, correlation, signal generation etc. is performed by software. This means SDRadar can be flexibly deployed for different purposes and with a relative short development process. In this paper, we present a generic SDRadar system that can operate in different setups for near-field monitoring applications. Practical solutions for traditional limitations in SDRadar and high sampling rates are introduced, and its performance is demonstrated using a commercial universal software radio peripheral (USRP) device with four synchronized receiving channels and a maximum sampling rate of 100MHz. Additionally, a GPU accelerator has been implemented to deal with the high sampling rate. Three different setups have been tested to demonstrate the feasibility of the propose SDRadar system with distributed nodes, vertically positioned nodes and a miniature scenario. Recorded Doppler signatures have shown the proposed SDRadar can effectively capture the body and hand gestures. Such results can be used in a range of applications such as eHealth, human-machine interaction and indoor tracking.**

***Index Terms*—Software Defined Radar, Doppler signature, multichannel, Real-Time, GPU accelerator**

I. Introduction

Radar's possibilities are very broad and cover different applications like long air surveillance, short field surveillance, target detection. These highly specialised radar systems have diverse demands but on the same platform (ship, aircraft, base station, Unmanned Aerial Vehicles (UAV) and others) [1]. Previous and current generations of radar system are generally highly specialized, built with Application Specific Integrated Circuit (ASIC) that are nearly impossible to reconfigure to other purpose [2]. In addition, there are growing number of other RF sensors, such as communication system, navigation system, IoT sensors etc. Constraint by the hardware platform, limited place and system cost, future radar system will thus have to present multifunction capabilities. Recently, a number of multi-function radar system have employed the idea of software-defined techniques. For example, weather radar [3] leverages SDRadar joint with Field Programmable Gate-Array (FPGA) for multichannel receiver.

Due to the similar requirement in hardware and high flexibility in software, SDRadar is normally implemented with commercial Software-Defined Radio (SDR) devices. The concept of SDR is for digital signal processing to be performed completely in software or FPGA with a RF-front end which includes analog-to-digital conversion (ADC) and digital-to-analog conversion (DAC), mixers and filters [4]. With appropriate design, commercial SDR devices can be configured as accessible and inexpensive platforms for SDRadar.

The universal software radio peripheral (USRP) family of SDR hardware platforms have been used widely for SDRadar which have been used across a number of remote sensing applications. For example, early work [5] configured SDR as a passive radar. This system demonstrated the feasibility of through wall detection by using a third-party signal. Another work [6] builds an ionospheric sounding radar, which includes a pair of synchronized transmitter and receiver. However, the low instantaneous bandwidth of the USRP device cannot provide sufficient range resolution when the high range resolution is critical for applications [7]. This can be improved by combining a series of sub-pulses to stimulate an Ultra-Wide-Bandwidth (UWB) radar [8] to enable higher range resolution. The work [9] further explores SDR-based UWB radar for landmine detection. Our previous work [10], [11] presented a Passive WiFi Radar (PWR) that used the WiFi AP as the signal source to illuminate the enviornment. The system was designed for real-time operation, however it was limited by single surveillance channel and relatively low sampling rate.

As a SDRadar, most of its signal processing are implemented in software. Because of the complex algorithms, higher sampling rate and more receiving channels, the requirement in computational power has been increased exponentially. For example, early member of USRP family [12] only has single channel with sampling rate of 20MHz, whereas new USRP device [13] has four channels with up to 100MHz. These additional channels are very important for SDRadars, they could be configured as multistatic radar [14] or phased array [15]. These channels imposes a significant amount of data. Thus, to enable real-time processing, these SDRadars need to be optimized to deal with high sampling rates.

Our previous work in this field [10] was a single channel SDRadar operating at a max sampling rate of 10MHz. In this paper, we report on advancements that allow our multichannel SDRadar to operate in real-time and with a significant improvement on sampling rate up to 80MHz. This is achieved by dividing a long serial processing is segmented into several short processing and switching to a parallel framework. The system architecture and signal processing methods to enable real-time multichannel process are described including batching the data to facilitate very high data throughput,

and three-level process design to ensure the balance between Data Acquisition (DAQ) and processing. Additionally, a GPU accelerator has been implemented to speed up the FFT process. The idea is to transfer part of signal processing onto GPU to save computation power in CPU for task like DAQ and data pre-processing. The overall SDRadar processing speed has observed a considerable improvement with the new architecture and GPU accelerator.

Comparing to previous works [5], [8], [10], the following contributions are made by this paper:

- This paper presents a robust high-speed multichannel SDRadar that capable of processing sampling rates up to 80MHz without dropping samples. This new SDRadar is 8 times faster than the system reported in our previous work [10].
- The proposed architecture offers flexibility and can be used for different SDR devices including a three-level system design for constant DAQ rate, an enhancement in measurement rate by using overlapping window and a parallel framework to break long processing into short processing.
- We implement a GPU accelerator to speed up the FFT calculation which is jointly designed with SDRadar. The GPU accelerator shows a acceleration ratio of 2-4 when compared to CPU-only process.

The rest of this article is organized as follows: Section II presents the concepts of proposed SDRadar and corresponding signal processing, in particular, the cross ambiguity function and direct signal suppression; Section III outlines the system architecture and GPU accelerator for high-speed processing; Results from three experiments, distributed node, vertical node and miniature scenario are presented in Section IV. Finally, the conclusions and potential applications are summarized in Section V.

II. Signal Processing for SDRadar

We consider the signal processing that can be applied to various signal source like FMCW, STCW etc, and passive signal source like WiFi AP and other third party transmitter. This is achieved by correlating the transmitted signal and received signal to calculate the relative time delay and Doppler shift. In such scenario, two channels are defined: reference channel $s_r(t)$ which recreate the transmitted signal and surveillance channel $s_s(t)$ which receive the signal from surveillance area. In the case of a multichannel SDRadar, there will be one reference channel and multiple surveillance channels.

We assume a perfect reference channel where the transmitted signal is measured directly from signal source as $s_r(t)$. The surveillance channel consists of both consists both direct signal and target reflections. These reflections are due to stationary clutters or moving objects that can be described by a summation of delayed and phase shifted transmitted signal. Received signal at surveillance channel $s_s(t)$ can be expressed as:

$$s_s(t) = \sum_p A_p e^{j2\pi f_d t} s_r(t-\tau) + n(t) \tag{1}$$

where p is the number of reflected paths, and A_p, τ, f_d are the attenuation factor, time delay, Doppler shift for pth path respectively. $n(t)$ represents the additive white Gaussian noise (AWGN).

In time domain and discrete domain, definition of cross-correlation can be written as Equation (2) and (3):

$$(s_s \cdot s_r) = \int_{-\infty}^{\infty} {s_r}^*(t) s_s(t+\tau) dt \tag{2}$$

$$(s_s \cdot s_r) = \sum_{n=-\infty}^{\infty} {s_r}^*[n] \times s_s[n+\tau] \tag{3}$$

where $(s_s \cdot s_r)$ denotes cross-correlation on surveillance and reference channel, and $*$ represents the complex conjugate. The cross-correlation implementation suggested by [16] can be defined as:

$$(s_s \cdot s_r) = IFFT(FFT(s_s^*) FFT(s_r)) \tag{4}$$

where FFT is fast Fourier transform and IFFT is inverse fast Fourier transform.

A. Cross-Ambiguity Function

Cross-Ambiguity Function (CAF) is a 2D range-Doppler matrix. It first performs a cross-correlation between the reference and surveillance channel as pointed in equation (4). This calculates the time delay (range information) between two signal. However, equation 4 cannot be directly used in SDRadar. Considering the sampling rate of 20MHz (a typical bandwidth of WiFi signal at 2.4GHz), which means there are 20M data points to be processed every second. Particularly, FFT processing over long sequence is very slow with CPU. This makes the cross-correlation almost impossible to be processed in real-time. Therefore, a batch process has been applied which segments a long sequence s into L short and equal length sequences as $s = [s^1, s^2, ..., s^L]$, each has a length of l_s. Cross-correlation with batch process [17] can be expressed as:

$$({s_s}^i \cdot {s_r}^i) = IFFT(FFT({s_s}^i)^* FFT({s_r}^i)), i = 1, 2, ..., L \tag{5}$$

where $({s_s}^i \cdot {s_r}^i)$ denotes the cross-correlation under batch process with a size of $L \times l_s$. Afterwards, Doppler shift can be calculated upon the time delays by performing another FFT over the short sequence. This would generate the CAF matrix $CAF(\tau, f_d)$ as:

$$CAF(\tau, f_d) = FFT({s_s}^i \cdot {s_r}^i)_B^T \tag{6}$$

where T represents the matrix transpose. Apply equation (1) into (6), the CAF calculation can be expressed as:

$$CAF(\tau, f_d) = \sum_{i=1}^{L} \int_0^{l_s} s_s^i(t) {s_r^i}^*(t-\tau) e^{j2\pi f_d t} dt \tag{7}$$

B. Direct Signal Cancellation

In real-world, surveillance channel not only receives Doppler-shifted target echoes but also consists of a direct signal interference (DSI) component, stationary clutter and thermal noise. Surveillance channel can be seen as a composition of linear combinations of the delayed signal. Expression of received signal in surveillance channel can be written as:

$$s_s(t) = s_{dsi}(t) + s_{clutter}(t) + s_{tar}(t) + n(t) \quad (8)$$

where $s_{dsi}(t)$ is the signal from direct path, $s_{clutter}(t)$ is the signal reflected from surrounding clutter and $s_{tar}(t)$ is the signal reflected from target. Thus the CAF matrix can be seen as the composition of linear combinations of the CAF matrix for the individual components of $s_s(t)$. We also note that, in Equation (8), only the $s_{tar}(t)$ is expected to have main peaks on nonzero Doppler bins. However, both DSI and clutter components can produce both range and Doppler sidelobe responses. Replacing equation (8) into (6), the cleaned CAF matrix [11] (with target echo only) can be extracted by removing other interference components as:

$$\begin{aligned} CAF_{tar}(\tau, f_d) =& CAF_(\tau, f_d) - CAF_{dsi}(\tau, f_d) \\ & - CAF_{clutter}(\tau, f_d) - n(t) \end{aligned} \quad (9)$$

where CAF_{dsi}, CAF_{tar}, $CAF_{clutter}$ are the CAF values arising from the direct signal, target echo and stationary clutter, respectively. To remove the DSI CAF_{DSI} and clutter $CAF_{clutter}$ components from CAF matrix, a shifted magnitude-scaled and phase-corrected version of CAF matrix was used as the reference. This is calculated by using a self-ambiguity surface which correlates with the reference signal itself.

$$CAF^k(\hat{\tau}, \hat{f_d}) = CAF^{k-1}(\tau, f_d) - \alpha^k CAF_{self}(\tau - T_k, f_d) \quad (10)$$

where α^k is the maximum absolute value of $CAF_{self}(\tau - T_k, f_d)$, T_k is the phase shift factor refers to the α^k. CAF_{self} is the self CAF over the reference channel. Afterwards, the phase of $CAF_{tar}(\tau, f_d)$ is shifted by multiplication with a complex phasor $e^{j\Delta\phi}$, where $\Delta\phi$ is the difference in phase between the peak in $CAF_{tar}(\tau, f_d)$ and the peak found along the zero Doppler bin.

III. System Design

A. System Overview

Fig 1 presents the flow chart of the SDRadar system. Firstly, a USRP-2945 has been used as the RF receiver which contains four synchronized channel (one for reference channel and three for surveillance channel). Since, the system does not transmit any signal, an external signal source is therefore needed. This can either an active source (FMCW signal) or a passive source (WiFi AP). Active source can be directly connected to the reference channel, while passive source can be wirelessly captured by using a directional antenna as discussed in [10]. The remaining three surveillance channels will be used for monitoring with different setup.

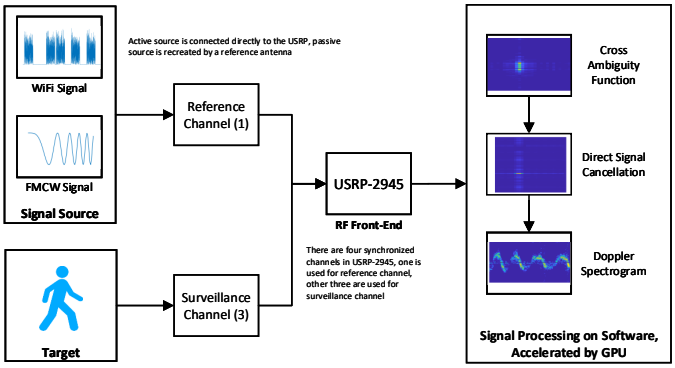

Fig. 1: Overview of proposed multichannel SDRadar system

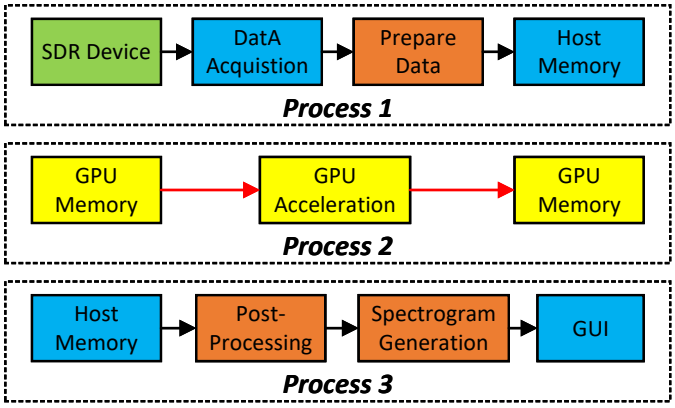

Fig. 2: Flow chart of processing

Afterwards, sampled data are passed to a computer and processed by cross ambiguity function and direct signal cancellation which are coded in LabVIEW. Additionally, a GPU-based accelerator has been implemented to speed up the processing by relieving the heavy computational load from CPU. The last step is to generate the Doppler spectrogram upon the cleaned CAF as indicated in Equation 10.

B. SDRadar Design

The proposed SDRadar is designed with three major tasks including raw data DAQ (Task1, T1), multichannel cross-correlation by GPU (Task2, T2) and spectrogram generation (Task3, T3), respectively. The flow chart of SDRadar is shown in Fig 2. Solid arrows indicate the the main data stream (in CPU), and red arrows indicate the data flow inside the GPU.

In T1, raw RF signal is acquired by the SDR device and transferred into the host computer's memory. Afterwards the raw signal is sorted into batches data. It is either saved into hard drive for off-line processing or downloaded into GPU memory through the PCIE X 16 interface for subsequent online processing. In T2, the GPU accelerator performs cross-correlation over the batch data with parallel processing. The obtained data are uploaded to host memory from GPU memory again in the end of process. In T3, Doppler spectrogram is generated by the CPU from the data loaded in the host memory

Algorithm 1 Implementation of Cross-correlation

Require: Sampling data from SDR devices

1: Reshape data into parallel batches on CPU
2: Download from CPU to GPU memory
3: Complex conjugate on received signal s_r
4: FFT on each batches on GPU
5: Multiplication between the transmitted signal s_t and received signal s_r
6: IFFT on each batches on GPU
7: Upload from GPU to CPU
8: Intercept first 30 samples within each batches
9: FFT across batches on CPU

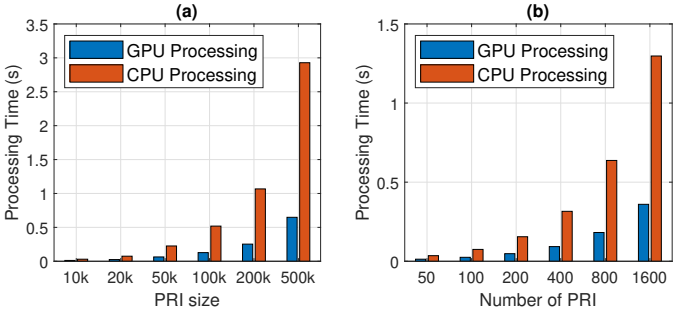

Fig. 3: Performance comparisons between CPU and GPU based cross-correlation processing: (a) PRI length and (b) number of PRI

for result display and data log for further processing like activity recognition, identification etc.

C. GPU Accelerator

Consider the architectures of GPUs and CPUs, CPU is designed to handle a wide-range of tasks whereas GPU is specifically designed for computationally intensive calculations that have high parallelism rates. For this reason, with more transistors, GPU is good for data processing rather than data caching and flow control. This difference in architecture determines the processing speed and is an essential factor for a SDRadar system. To deal with the multichannel aligned with high sampling rate, we implement a GPU accelerator to speed up the CAF process. Recall the cross-correlation in Equation 4, the detailed description about GPU implementation is shown in Algorithm 1.

Fig 3 shows the performance comparison results between the CPU and GPU based cross-correlation processing upon various data sizes. The PRI size needs to be larger to capture effective signals,for example in the case of passive sensing with WiFi signal, where PRI number defines the maximum detectable velocity. As it can be seen, the processing time of GPU is much shorter than CPU for any values of PRI and its length, even through the CPU-based processing has been optimized with matrix processing. In general, GPU-based processing shows 2 to 5 times faster than the CPU-based processing. There are two main factors: 1. faster FFT processing over longer data in each block; 2. more efficient processing due to the highly paralleled core hierarchy in GPU.

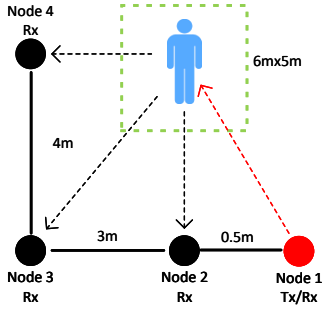

Fig. 4: Experiment setup 1: distributed nodes

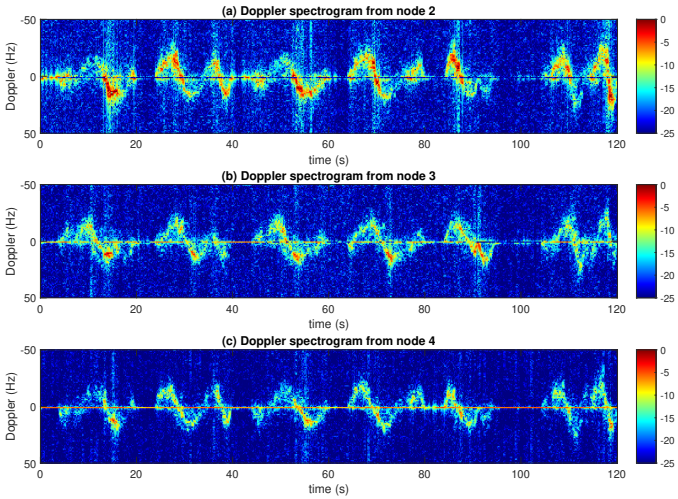

Fig. 5: Doppler spectrogram captured by 3 nodes (random walking)

IV. Experiment Results

Three experiments were designed to verify the effectiveness of the proposed SDRadar system, one for distributed nodes at various aspect angles to the target, one for vertical nodes to capture the signal at different height and one for hand gesture. In the experiments, the Doppler shifts that are caused by human during the are detected and recorded in real-time. The experimental setup and results are described in detail in this section.

A. Distributed Node

Experiment setup for distributed node is shown in Fig 4. A WiFi AP was used as the signal source at node 1 together with the reference channel. The WiFi was fixed at channel 149 (5.745GHz/20MHz) with a frame rate of 200Hz. Node 2,3,4 were separated at different angles towards the monitoring area to captured Doppler signatures from different angles. Since this is a line-of-sight scenario, four low gain (6 dBi) direction antennas were used. The CPI was set at 1s, PRI was 10ms and 100 Doppler bins.

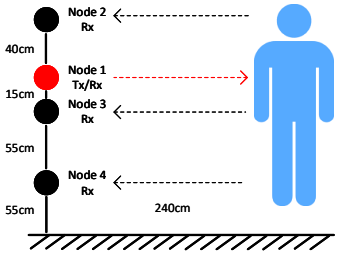

Fig. 6: Experiment setup 2: vertical nodes

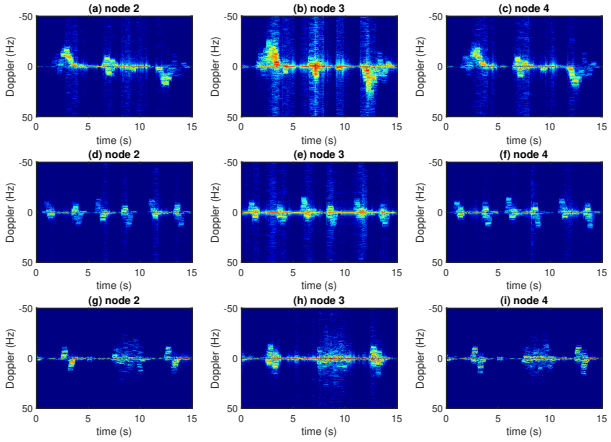

Fig. 7: Doppler spectrogram captured by vertical node: (a)(b)(c) walk-turn-walk; (d)(e)(f) stand-sit-stand (three times); (g)(h)(i) (pick up)-(body rotate)-(pick up)

In this experiment, a person was asked to walk around randomly within an area of 6m×5m with pauses during the experiment. This is to demonstrate the sensitivity of the SDRadar. Figure 5 presents the detection results captured by all 3 nodes. It can be seen that the real-time Doppler record for each period of walking can be distinguished from the others. Doppler signatures are varied in each node due to the variations in monitoring angle. This difference provided by spatial diversity is very important for many machine learning tasks to improve their accuracy like activity recognition, localization, people counting etc. Additionally, unlike the CSI-based systems [18], the phase noise of our SDRadar exhibits better stability and can more easily be extracted to provide an indication of target direction, where the positive pulses represent the person moving towards the antenna and negative pulses represent away from the antenna. Moreover, we can also observe the micro-Doppler caused by limbs movement, which indicates the system is very sensitive even for small movements. Such spectrogram provides rich information about the monitored target, the value of Doppler shifts is related to the particular geometry of transmitters and receivers and can be resolved by the bistatic radar equations.

B. Vertical Node

Vertical node is used to capture the Doppler signature at different height. This configuration can be used for person identification to provide diverse information from different part of body. Fig 6 shows the experiment setup for vertical node, where all nodes were placed vertically separated at 55cm, 110cm, 125cm and 165cm above the ground respectively. The CPI was set at 1s, PRI was 10ms and 100 Doppler bins.

In this experiment, a person was asked to perform 3 activities including walking, sitting/standing, picking and body rotating at a distance of 240cm away from the antennas. The collected spectrograms are shown in Fig 7. We can see clear difference in Doppler signatures among different nodes. For example node 3 gave the strongest Doppler signature comparing to node 2 and node 4. This is because node 3 was pointing to the torso of the human body which has larger reflection area than node 2 (head) and node 4 (legs). However, each nodes also capture signal from other parts of body, this makes the Doppler spectrogram from each node has similar shape. Walking spectrogram has the largest Doppler shift as expected for its fast velocity. In comparison, activities like sitting/standing are with much lower Doppler shifts due to relative small velocity. Specially the body rotating (middle part in Fig 7(g)(h)(i)) has the lowest Doppler shifts because the torso does not move, however there are micro-Doppler shifts in both positive and negative domains for the torso rotation. By studying the difference among different nodes, we can easily apply them to AI applications like human identification, crowd counting, etc.

C. Miniature Scenario

Many electronic devices such as laptops, tablets and mobile phone can be integrated with such system with self WiFi signal or active signal. To demonstrate this concept, we placed the transmitter and receiver to setup a miniature scenario to capture hand movements in front of a small area. The experimental setup is shown in Fig 8, where all four nodes were placed in a square manner at a distance of 40cm×30cm. To reflect the small hand movement, we reduce the CPI to 0.5s, PRI to 0.25s and 200 Doppler bins.

In this experiment, we carried out four typical types of hand gestures to demonstrate the performance of proposed SDRadar over short distances. The corresponding micro-Doppler records for hand motions are shown in Fig 9. The examples used in this experiment were forward-backward, left-right, clockwise and counterclockwise movements. The micro-Doppler signatures show clear periodical characteristics. Different hand movements generated different cyclical patterns that are clearly distinguishable. Additionally, there are also clear differences within each node. For example, in forward and backward movement (first column), the micro Doppler in node 2 is stronger than that in node 3 and node 4, as the result of monostatic layout in node 2. In comparison, left-right movement (second column) shows that micro Doppler node 3 and 4 is stronger than node 2. This is because the bistatic velocity of left-right movement is almost zero towards node 2.

These preliminary results clearly show that the proposed SDRadar system has the ability to record the micro-Doppler trigger by the hand gesture. Such results can be used for

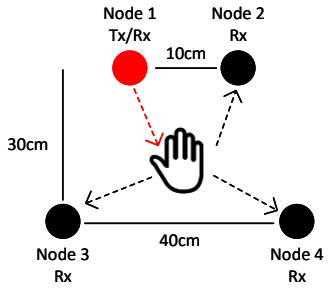

Fig. 8: Experiment setup 3: miniature setup

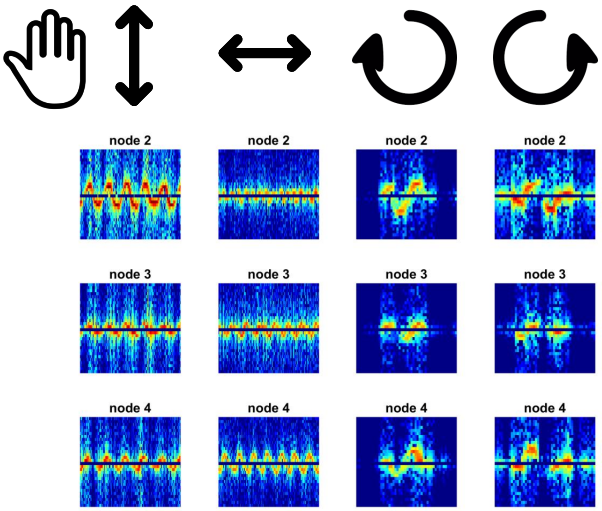

Fig. 9: micro-Doppler captured by 3 nodes (hand gesture)

hand gesture recognition in the same way as the body gesture in Section IV A and B. Moreover, benefit by the multiple node surveillance, the diversity gain can further improve the reliability and sensitivity for hand gesture.

V. Conclusions

In this paper, we present a high-speed SDRadar for multi-purposes in near-field sensing applications. We have demonstrated the system design and implementation including the overall system architecture, signal processing for SDRadar and GPU-based accelerator. Three experiments were carried to verify these principles including distributed node, vertical node and miniature scenario. Real-time body and hand gestures have been proved the Doppler signature collection work effectively with the proposed system. The multistatic setup can further benefit the SDRadar by collecting Doppler information from aspect angles with geometry diversity.

Potential applications are widespread for this system, including motion sensor for many scenarios for example human machine interaction, healthcare, security etc. Future works include further experiments for more classes of body and hand gestures for classification purposes. Additionally, various signal processing will be investigated in order to improve the capability of the system focus on improved interference suppression and sensitivity.

Acknowledgments

This work is part of the OPERA project funded by the UK Engineering and Physical Sciences Research Council (EPSRC), Grant No: EP/R018677/1.

References

[1] T. Debatty, "Software defined radar a state of the art," in *2010 2nd International Workshop on Cognitive Information Processing*. IEEE, 2010, pp. 253–257.

[2] D. J. Fouts, P. E. Pace, C. Karow, and S. R. Ekestorm, "A single-chip false target radar image generator for countering wideband imaging radars," *IEEE Journal of solid-state circuits*, vol. 37, no. 6, pp. 751–759, 2002.

[3] B. L. Cheong, R. Kelley, R. D. Palmer, Y. Zhang, M. Yeary, and T.-Y. Yu, "Px-1000: A solid-state polarimetric x-band weather radar and time–frequency multiplexed waveform for blind range mitigation," *IEEE Transactions on Instrumentation and Measurement*, vol. 62, no. 11, pp. 3064–3072, 2013.

[4] F. Harris and W. Lowdermilk, "Software defined radio: Part 22 in a series of tutorials on instrumentation and measurement," *IEEE instrumentation & measurement magazine*, vol. 13, no. 1, pp. 23–32, 2010.

[5] B. Tan, K. Woodbridge, and K. Chetty, "A real-time high resolution passive wifi doppler-radar and its applications," in *2014 International Radar Conference*. IEEE, 2014, pp. 1–6.

[6] Z. Zhao, M. Yao, X. Deng, K. Yuan, H. Li, and Z. Wang, "A novel ionospheric sounding radar based on usrp," *IEEE geoscience and remote sensing letters*, vol. 14, no. 10, pp. 1800–1804, 2017.

[7] Y.-K. Kwag, I.-S. Woo, H.-Y. Kwak, and Y.-H. Jung, "Multi-mode sdr radar platform for small air-vehicle drone detection," in *2016 CIE International Conference on Radar (RADAR)*. IEEE, 2016, pp. 1–4.

[8] S. Prager, T. Thrivikraman, M. S. Haynes, J. Stang, D. Hawkins, and M. Moghaddam, "Ultrawideband synthesis for high-range-resolution software-defined radar," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 6, pp. 3789–3803, 2019.

[9] S. Prager and M. Moghaddam, "Application of ultra-wideband synthesis in software defined radar for uav-based landmine detection," in *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2019, pp. 10 115–10 118.

[10] W. Li, R. J. Piechocki, K. Woodbridge, C. Tang, and K. Chetty, "Passive wifi radar for human sensing using a stand-alone access point," *IEEE Transactions on Geoscience and Remote Sensing*, 2020.

[11] W. Li, R. J. Piechocki, K. Woodbridge, and K. Chetty, "Physical activity sensing via stand-alone wifi device," in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[12] Ni usrp 2920. [Online]. Available: https://www.ni.com/en-gb/shop/hardware/products/usrp-software-defined-radio-device.html

[13] Ni usrp 2945. [Online]. Available: https://www.ni.com/en-gb/support/model.usrp-2945.html

[14] Z. Chen, G. Li, F. Fioranelli, and H. Griffiths, "Personnel recognition and gait classification based on multistatic micro-doppler signatures using deep convolutional neural networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 5, pp. 669–673, 2018.

[15] S. A. Alawsh, O. A. Al Khazragi, A. H. Muqaibel, and S. N. Al-Ghadhban, "Sparse direction of arrival estimation using sparse arrays based on software-defined-radio platform," in *2017 10th International Conference on Electrical and Electronics Engineering (ELECO)*. IEEE, 2017, pp. 671–675.

[16] Cross-correlation theorem. [Online]. Available: https://mathworld.wolfram.com/CrossCorrelationTheorem.html

[17] W. Li, B. Tan, and R. Piechocki, "Passive radar for opportunistic monitoring in e-health applications," *IEEE journal of translational engineering in health and medicine*, vol. 6, pp. 1–10, 2018.

[18] Y. Ma, G. Zhou, and S. Wang, "Wifi sensing with channel state information: A survey," *ACM Computing Surveys (CSUR)*, vol. 52, no. 3, p. 46, 2019.