

TOPICAL REVIEW

Scattering Centers to Point Clouds: A Review of mmWave Radars for Non-Radar-Engineers

HARRY D. MAFUKIDZE^{1,2}, AMIT K. MISHRA², (Senior Member, IEEE),
JAN PIDANIC³, (Senior Member, IEEE), AND SCHONKEN W. P. FRANCOIS²

¹Department of Applied Physics and Telecommunications, Faculty of Science and Technology, Midlands State University, Gweru, Zimbabwe

²Electrical Engineering Department, University of Cape Town, Cape Town 7700, South Africa

³Department of Electrical Engineering, Faculty of Electrical Engineering and Informatics, University of Pardubice, 532 10 Pardubice, Czech Republic

Corresponding author: Jan Pidanic (jan.pidanic@upce.cz)

The work has been supported by the programme INTER-EXCELLENCE (LTAIN19100) funds of the Ministry of Education, Youth and Sports, Czech Republic. Project LTAIN19100 “Artificial Intelligence Enabled Smart Contactless Technology Development for Smart Fencing”.

ABSTRACT Recently, mmWave radars have been gaining popularity, thanks to their low cost, ease of use and high-resolution sensing. In this paper, we provide a review of the mmWave radar data processing frameworks, starting from mathematical foundations to applications. Specifically, we focus on the mmWave radar point cloud as a robust data structure representing compressed signatures for target recognition and classification. We first focus on the generation of the radar point clouds, and the signal processing algorithms designed for their unique characteristics. Then, we illustrate how the radar point clouds are prepared for feature extraction and classification using machine learning and deep learning approaches. Finally, we summarize the state-of-the-art applications, open datasets, developments and future research directions in this field.

INDEX TERMS mmWave radar, single-chip radar, radar point clouds, scattering centers, COVNETs.

I. INTRODUCTION

Automotive radars have seen a lot of research, innovation and development over the last decade. A recent report on the automotive radar market projects the value to reach \$10 billion by 2028 [1]. This rapid rise has occurred due to the increased production surrounding autonomous driving, resulting in a series of enabling standards [2]. In particular, the technical and operational characteristics of automotive radars have facilitated the development of a new type of radar. This state-of-the-art radar required a break from the conventional way of building and packaging radar systems.

Consequently, single-chip radar systems operating at mmWave frequencies, have become available to the wider research community. Established semiconductor manufacturing companies such as Texas Instruments (TI), Infineon, NXP and Analog Devices are at the forefront of this innovation. Their solution is a single-chip complementary metal-oxide-semiconductor (CMOS) radar sensor that integrates the

The associate editor coordinating the review of this manuscript and approving it for publication was Cheng Hu¹.

entire radar processing chain on a single chip. The need for additional external circuitry has thus been eliminated [3]. As we can expect, there are many benefits from such self-contained, single-chip radar devices. Examples include low cost, low power consumption, high spatial and velocity resolution, and low-physical size of the radar unit. Such benefits are ideally suited for driver assistance systems (ADAS) such as adaptive cruise control, autonomous emergency braking, blind spot detection, forward collision warning system, intelligent park assist, pedestrian and animal detection among others [1], [4].

Following the availability of inexpensive and high-performance mmWave radars, there have been a number of works in the open literature using mmWave radars for a range of new and emerging applications. Examples include consumer electronics [5], the Internet of Things (IoT) [6], industrial applications [7], driver assistance systems [4], human gait monitoring [8], healthcare [9], and posture estimation [10]. The majority of these works concentrate on employing artificially intelligent sub-systems in their data processing pipelines. This has resulted in faster, more

reliable, and more accurate decision-making due to automated identification, recognition, and classification of radar targets. In this respect, mmWave radars have shown great potential to replace various invasive and bulky sensors that have been predominantly used on the same tasks.

Given the high resolution of mmWave radars, the scattering centers in mmWave radar data are numerous and resemble the scattering center behavior from other high-resolution coherent imaging systems (like LiDAR and laser). Following the nomenclature in those domains, scattering centers in mmWave data are often called point clouds. mmWave radars have also created an ecosystem where innovators, who are not experts in radar, are using mmWave radars to solve a range of interesting and novel challenges. In spite of the surge in projects using mmWave radars, a good review of these radars, specifically written for non-radar engineers, is scarce. In this paper, we aim to address this gap by writing a review of mmWave radars. We discuss radar scattering mechanisms and how scattering centers can be modelled by a point-cloud model. Additionally, this paper serves as a reference guide for developers who want to build cutting-edge radar applications but do not have access to data collection equipment. We summarize the key contributions of our work as follows:

- 1) We provide an up-to-date review of modern developments in mmWave radar applications. Firstly, we select works that rely on radar point clouds, then we study how the application's algorithms processes the point clouds. Next, we assess the performance of each point cloud processing pipeline on deep learning models, given that most application areas in the literature use deep learning algorithms for classification.
- 2) We discuss the strengths, weaknesses, and directions of further research on the application of point clouds in various mmWave radar application areas.
- 3) We review ten public datasets for various scenarios collected using various automotive mmWave radars. The datasets could be used to benchmark target detection, recognition and classification algorithms.

The rest of the paper is organized as follows. The rest of the paper is organized as follows. Section II gives an overview of mmWave radars, and the current research trends. We discuss the scattering center model of radar targets in Section III. In Section IV, we define the radar point cloud, and describe its mathematical formulation. We discuss the algorithms that have been used to process them, as well as challenges associated with the radar point cloud model. Additionally, we present point cloud feature extraction techniques, along with the methods of transforming the point cloud data. In Section V, we present the various application areas of mmWave radars, as well as issues for further research. Section VI reviews the deep learning models that have been used to process the radar point clouds and finally, Section VII concludes the paper. In Figure 1, we show a diagrammatic view of this survey.

II. mmWave RADAR SENSORS: AN OVERVIEW

Radar systems are typically constructed by assembling several individual blocks such as antenna hardware, radiofrequency transmit and receive chains, amplifiers, analog to digital converters, and digital signal processors, among others. This has been the only way of producing radar units for several years since their inception in the first world war [11]. Until recently, companies such as Texas Instruments, Infineon, NXP and Analog Devices have challenged this paradigm. One of the major vendors in the field of single-chip radar products is Texas Instruments (TI). The company has produced a wide range of high-performance single-chip mmWave radar sensors to accelerate development in automotive (AWR1xxx family) or next-generation industrial applications (IWR1xxx family). TI's product portfolio consists of a large number of high-performance mmWave radar sensors that provide accurate and high-resolution sensing. To date, the catalog consists of AWR2944, AWR1843AOP, AWR6843AOP, IWR6843AOP, AWR6443, AWR2243, AWR6843, IWR6443, IWR6843, IWR1843, AWR1843, AWR1243, AWR1443, IWR1443, AWR1642 and IWR1642. Despite the difference in the chip-set architecture of TI's mmWave products, they operate in the extremely high-frequency spectrum between 60GHz – 81GHz and they all use the frequency modulated continuous wave (FMCW) radar principle for object detection and ranging.

Another semiconductor manufacturer in this segment is Infineon Technologies. The company produces a wide range of high-performance, single-chip radar sensors for different applications. Their product portfolio covers the 24GHz (BGT24), 60GHz (BGT60) and 77/79GHz ISM bands, with potential uses in vital signs monitoring [12], motion detection [13], direction estimation, distance, and speed measurement [14]. Just like any single-chip radar solution, Infineon's product offerings have a small form factor, integrated antennas, greater accuracy, low power consumption and cost. Above all, they are compatible with Arduino, one of the most popular electronics prototyping platforms. This allowed non-radar engineers to implement cutting-edge radar solutions in a familiar development environment.

In the literature, several authors have used Infineon's mmWave radars for short-range (2m), medium-range (25m), and long-range (250m) radar applications. In particular, their high-resolution sensing capabilities make them ideal in applications such as vital signs monitoring [15], air-writing system [16], human detection [17], hand gesture recognition [18], and human activity recognition [19], [20], [21]. Unlike other product offerings in this category, Infineon's radar sensors use a variety of software interfaces to extract the raw I/Q data from the front-end sensor. This simplifies the development of custom radar signal processing steps.

However, using single-chip radar platforms for target detection, recognition and classification has certain operating limitations. Notably, mmWave signals suffer from high propagation attenuation [22]. This means that such sensors

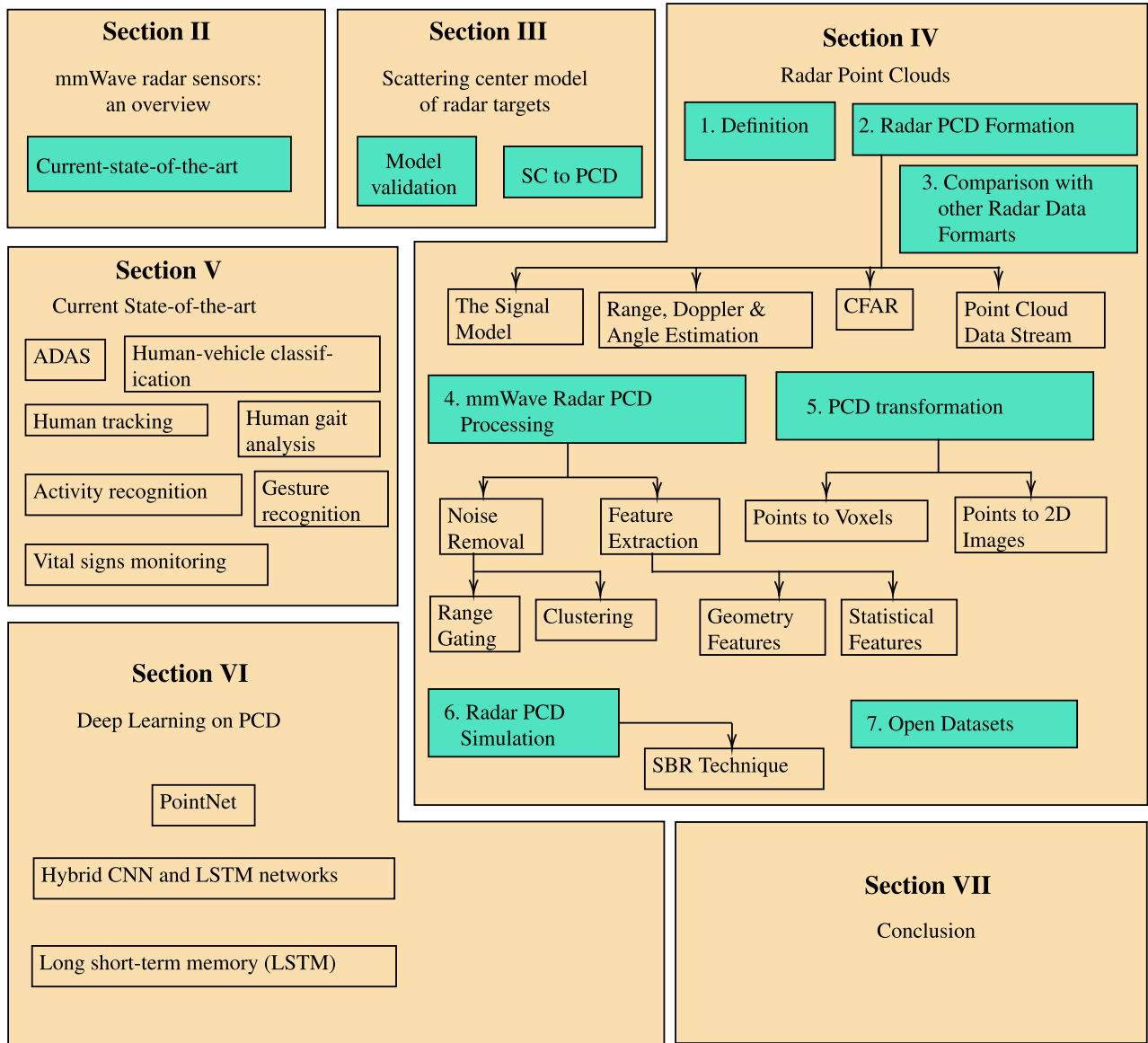


FIGURE 1. Survey organization.

are ineffective for very long-range operations ($>500m$). The majority of mmWave applications in the literature are thus confined to distances less than 20m from the radar. For example, mmPose: 5m [10] and milliEgo: 10m [23]. This is significantly lower than most operational commercial radar target detection applications.

Another weakness of single-chip radars arises from transferring data from the sensor’s internal memory to the host computer through UART [23]. Taking the AWR1642 sensor IC, for instance, the device has three levels of onboard memory with a total capacity of 1.5MB. The primary storage of the radar cubed data can only store 738KB worth of complex data, which is equivalent to a frame of received data. The transfer of data from this memory block happens during a period between two consecutive frames. This means that the size of the data that can be transferred during this period is limited by the number of frames per second of the chip

configuration [24], [25]. The workaround to this limitation is to implement some of the signal processing on the onboard C674x signal processor to reduce the size of the data that can be transferred through the USB port [8]. TI’s mmWave devices implement the low-level radar signal processing algorithms on the DSP to compute the Point Cloud Data (PCD) of the detected objects. This dataset is reduced in size since it is made up of only the $x(m)$, $y(m)$, $z(m)$ coordinates, Doppler information, time index and the power levels of each detected object. Moreover, this data representation format does not need long integration times to generate features for classification, thus making them favourable for real-time application scenarios.

A. CURRENT STATE-OF-THE-ART

The recent introduction of single-chip mmWave radars, and its subsequent success, have resulted in a tremendous

research effort to apply this relatively new technology in areas not previously envisaged. At the forefront is research related to automotive safety, driver assistant technology, vital signs detection and vehicle occupancy detection [4], [7], [8], [22], [26]. In addition to automotive safety, some researchers have pursued the development of post-processing, machine learning and deep learning algorithms on human activity recognition [27], people detection and tracking [28] and multiple patients behaviour detection [29]. Generally, this is considered a new field, and research in this field is still in its infancy, considering that many published manuscripts are from 2018.

One of the earliest reports on the use of automotive mmWave radars in the open literature was the one by Texas Instruments in 2017 [24], [25]. Later that year, Cesar et al presented the fundamentals of millimetre wave sensors in the company's whitepaper [25]. They introduced a fully contained, high-performance front-end radar product portfolio for modern automotive and industrial applications. In addition, the authors presented the fundamental concepts that govern the operation of their sensors, along with software toolkits for interacting with the radars. In that same year, Karthik et al also presented the AWR1243 radar as a potential front-end sensor for advanced driver assistance systems (ADAS) and autonomous driving safety systems [24]. Since then, we have seen several articles that showcase the many application areas of these sensors in the literature. Similarly, we have also witnessed an increase in the number of review articles that summarize modern developments in this field, especially in human activity recognition systems. Despite increased attention in mmWave radars, existing surveys focus on developments in the different application areas separately, for example, radar fall prevention [30], vital signs detection [31] and hand gesture recognition [32]. In contrast, only a small number of reviews cover the entire spectrum of mmWave radar application areas, for example, deep learning on 3D point clouds [33].

III. SCATTERING CENTER MODEL OF RADAR TARGETS

In modern mmWave automotive radar applications, the term "radar point clouds" is preferred to the original nomenclature "radar scattering centers". However, this preference is not clear, since the two terms both refer to the dominant scattering points of a radar target. Moreover, the definition of the point cloud is borrowed from other fields such as LIDAR, and adapted to radar. It is probable that this was to appeal to non-radar professionals who work with mmWave radar sensors without actually knowing the foundations of electromagnetic scattering. So far, there have been no attempts to define the radar point clouds in the context of radio wave propagation. In this Section, we discuss the radar scattering mechanisms, and the techniques to validate the scattering center models.

In Synthetic Aperture Radar (SAR) and Inverse Synthetic Aperture Radar (ISAR) imaging, the term *scattering centers* is often used to describe the location or the attributes

TABLE 1. α parameters for canonical scattering mechanisms [41].

α	Scattering geometries
1	Flat plate at broadside; dihedral
$\frac{1}{2}$	Singly curved surface reflection
0	Point scatterer; doubly curved surface reflection; straight edge specular
$-\frac{1}{2}$	Curved edge diffraction
1	Corner diffraction

of the dominant radar reflectors or scatterers on the target [34]. Therefore, this means that the radar response of a complex target originates mainly from these individual scattering points [34]. Each scattering center provides information such as location (x_k, y_k) , frequency dependence (α_k) , amplitude (A_k) , and polarimetric information. These attributes can then be used to give a physical description of the target, as reviewed by Moses et. al [35], where most of the feature extraction techniques are provided. The concept of the *scattering center* is not new, it dates back to the early development of the radar. The earliest work that relies on this concept can be traced to the introduction of SAR in the 1950s, where it was used primarily for military reconnaissance [36]. Later, it was used by NASA for oceanography missions in the 1970s, but the majority of works in the literature use this concept to create a radar signature database for automatic target recognition [37]. A much more detailed overview can be found in [38], [39], and [40].

From the classical radar literature, it is well-known that if the wavelength of the incident electromagnetic wave is smaller than the target extent, then the backscattered field is obtained by summing the contributions from the individual scattering centers [41]. This concept is well defined by the Geometric Theory of Diffraction (GTD) [43], which was discussed by Keller in 1962. From this theory, the Point Scatter Model in Equation 1 can be derived [41].

$$E(f, \theta) = \sum_{k=1}^K A_k \left(\frac{jf}{f_c} \right)^{\alpha_k} e^{\beta_k \theta} \cdot \exp \left\{ \frac{-j4\pi f}{c} (x_k \cos \theta + y_k \sin \theta) \right\} \quad (1)$$

Equation 1 represents a single polarization model for the backscattered electric field when the wavelength of the incident EM wave is smaller than the target extent. Here, $E(f, \theta)$ is the backscattered field as a function of frequency f and aspect angle θ . A_k is the magnitude and phase, x_k, y_k is the range and cross-range respectively. $\alpha_k \in [-1, -\frac{1}{2}, 0, \frac{1}{2}, 1]$ represent the target geometry as shown in Table 1, and β_k represents the aspect dependence of the k^{th} scattering center. Finally, solving Equation 1 produces a feature vector Θ in Equation 2, that can be used for target recognition:

$$\Theta = \{A_k, x_k, y_k, \alpha_k, \beta_k\}. \quad (2)$$

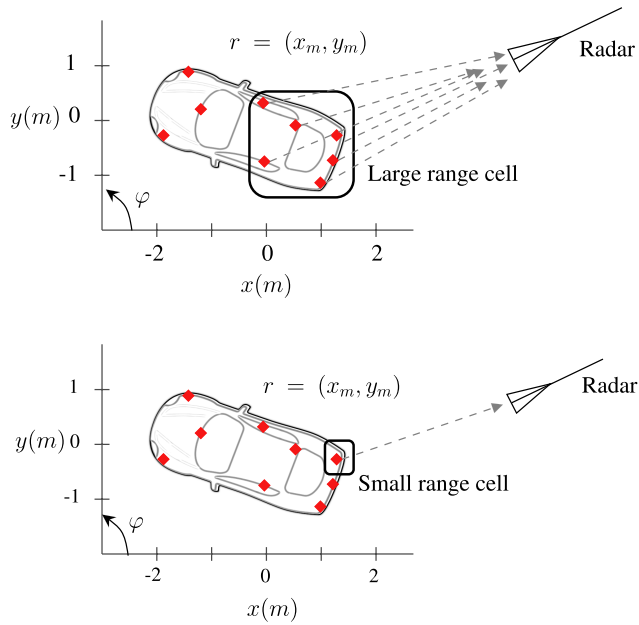


FIGURE 2. Illustration of the hypothetical scattering centers (represented by red diamonds) of a target vehicle. Suppose the vehicle has M scattering centers. It is thought that each scattering center can be due to different scattering mechanisms such as triple scattering, edge diffraction, travelling wave, specular scattering, and double scattering among others [42]. Here, $r = (x_m, y_m)$ is the range and cross-range of the m^{th} scattering center respectively. The backscattered field will be obtained by summing the contributions from the M scattering centers in the range/cross-range cell. The corresponding effect of the range/cross-range cell size on the amplitude and the number of scattering centers can also be seen.

Michael et. al in their work, “Feature Extraction Using Attributed Scattering Center Models on SAR Imagery”, proceed further to provide mathematical techniques for extracting each parameter of Θ [115]. Other variations of the scattering center model include attributes such as the length L_k of the canonical scatterer [44].

The diagram in Figure 2 is provided as a hypothetical example that illustrates the scattering centers of a vehicle radar target in a high-frequency radar operation. The scattering centers at each azimuth φ and elevation θ originate from the discrete irregular features or discontinuities on the surface of the target, such as edges, vertexes, curved surfaces, or cavities, among others [45]. The radar echoes from these scattering centers can be modelled by canonical scattering mechanisms that include, corner diffraction, edge diffraction, reflection from a sphere, reflection from a cylinder and lastly, reflection from a flat plate [46].

It can also be observed in Figure 2 that the number of scattering centers that can be observed in any range gate is greatly affected by the frequency bandwidth of the radar signal. In essence, a larger bandwidth improves the range resolution, and thus improves the radar’s ability to separate targets that are in close proximity [47]. For example, a 4GHz bandwidth of a mmWave automotive radar will have a range cell size of $\approx 4\text{cm}$, compared with a resolution of $\approx 1.5\text{m}$ that could be achieved with a bandwidth of 100MHz. This means

that for small wavelengths, each feature on the target will act as a scattering center, where its position is visible across multiple aspect angles. On the other hand, in low frequencies, a range gate will typically contain several scattering centers. The resulting scattering centers are the hypothetical centers where the phases are located due to random interference of reflections from these scattering centers.

A. SCATTERING CENTER MODEL VALIDATION

The scattering center models of radar targets need to be validated either experimentally or through numerical simulations to ascertain the accuracy and performance of the model. The first approach deals with conducting experiments in the field or in anechoic chambers. These experimental techniques estimate the backscattered radar response of realistic targets placed on measurement rigs. Such procedures are well documented in the literature. However, in the absence of experimental resources, electromagnetic (EM) simulation software is often applied to model electromagnetic scattering and wave propagation. Thus, the second approach consists of using numerical simulation tools that implement Maxwell’s equations - a set of equations which describe the basics of electric and magnetic fields [48]. They have proven to concisely solve EM-related problems. Therefore, computational programs that rely on this set of equations can be applied to provide EM solutions to geometrically complex targets or even electrically large targets [41], [49], [50], [51], [52].

Several established numerical simulation approaches for extracting the scattering centers of a target include the shooting and bouncing ray technique [49], and the uniform theory of diffraction (UTD) [41]. These methods fall into a class of asymptotic ray-tracing techniques that predict the EM scattering from 3D CAD models of real targets. Other techniques, using the full-wave numerical approach, include the multilevel fast multipole (MLFMM) method [50], method of moments (MoM) [51], and finite element method [52]. An exhaustive study of this field of computational electromagnetics has been conducted by [53] and validated by measurements [54].

Ultimately, model validation (either simulations or experimentally), determines the degree of closeness of the estimated and measured SC parameters. In other words, this is to check whether the scattering center positions predicted by the point scatter model, actually exist at those positions on the real target. If the calculated mean squared error is relatively small or if the results agree, then the SC model accurately represents the scattering behavior of radar targets.

B. FROM SCATTERING CENTERS TO mmWave RADAR POINT CLOUDS

In the collected literature, Radar Point Cloud is a term universally accepted to describe a list of detected objects returned by the radar processing chain. By applying a range FFT on the samples, a range-slow time matrix is obtained. An additional FFT is then applied to return a 2D range-Doppler matrix, followed by a CFAR operation to extract the range bin that

contains reflections from the targets. With mmWave radar sensors, the improved range/cross-range resolution enables the detection layer to return multiple reflection points from the scattering structure, giving a wide array of detected objects. Each detected object represents a scattering point with basic features such as range, cross-range and Doppler. Additional information such as the peak value and elevation (*in case of radars with etched antenna designs that allow elevation angle measurement*) can be extracted to reveal additional characteristics about the detected object.

Peak Grouping is also available as an option to reduce the number of detected points. Basically, this works by comparing the peak values of the neighboring bins in the detection matrix, and returning only those CFAR detected peaks with maximum values. Lastly, the algorithm outputs an array of the detected objects $\hat{\Theta}$, that represent the dominant scattering points of the target. A detailed mathematical formulation of the point cloud data is given in Section IV-B.

$$\hat{\Theta} = \{x_k, y_k, z_k, v, \text{peak_val}\}. \quad (3)$$

Already, we see some similarities between the Point Cloud description, $\hat{\Theta}$ in Equation 3, and the scattering center model, Θ in Equation 2. Mainly, both models provide a rich set of attributes for automatic target recognition, and they are both used for target signature compression. However, the main difference between the two models is that the scattering center model provides additional attributes that represent the geometry of the scattering structure, while the radar point cloud model fails to account for the orientation and geometry of the scatterer. Despite this notable difference, both models provide features that have been successful in a wide range of applications, from consumer electronics to military deployment.

IV. REPRESENTATION OF SCATTERING CENTERS AS POINT CLOUDS

At mmWave, much smaller physical features on the surface of targets act as phase centers to create scattering centers in the radar data. This makes it easy to extract the overall physical shape of the targets. With so many scattering centers in the signal, it is no longer practical to analyze each scattering center. Rather, clusters of scattering centers can be treated as features. Hence, researchers have been using the nomenclature from the image processing literature and calling clusters of scattering centers, extracted from mmWave radars, as a point cloud. It can be noted here that the detailed representation of physical features with point clouds comes at the cost of lost phase-center-based information extraction.

From this section onward, we shall refer to the scattering centers as radar point clouds, or just “point clouds”, to maintain consistency in the literature. Also, we refer to the point cloud as data points returned by the mmWave radar front end.

A. DEFINITION

The term “Point Cloud Data” was conceived to represent multidimensional data points from depth sensors such as

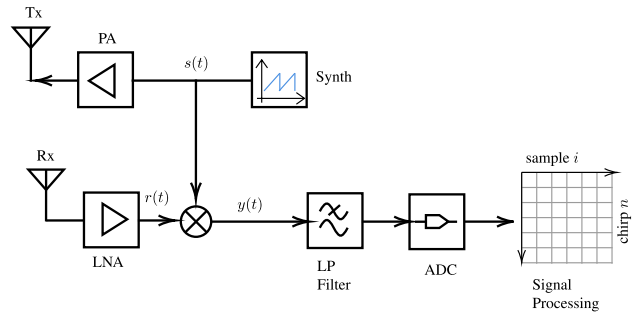


FIGURE 3. Basic components of the frequency-modulated continuous wave (FMCW) radar consists of a synthesizer, power amplifier (PA), transmit antenna, receive antennas, low-noise amplifier (LNA), frequency mixer, low-pass filter (LPF), an analog-to-digital converter (ADC), and signal processing chain.

LIDAR and range cameras. Some parts of the literature define point cloud data as an information model that is flexible and often used to compress the signatures of an object. In essence, the point cloud data comprises multiple frames of individual points that occupy unique locations in the Euclidean space \mathbb{R}^x , where x represents the dimensions of the data frame. We can therefore represent a point cloud of N points as $P = \{p_1, p_2, \dots, p_N\}$, where $p_n \in \mathbb{R}^x$ and $n = 1, \dots, N$ represents the dimensions of the point cloud, such that;

$$p_i = \begin{bmatrix} x_i \\ y_i \\ z_i \\ v_i \\ I_i \end{bmatrix} \quad (4)$$

From this representation, we can easily derive the object’s spatial (x, y, z) coordinates, along with additional information such as the speed (v) or intensity (I) of the reflected points. One of the main strengths of this representation is that point clouds convey vital information about the object while minimizing the computational and memory requirements. This makes it ideal for resource-constrained devices such as the TI mmWave radar. Besides reduced computational requirements, the point clouds represent the signatures of a target in point form, making it possible to represent complex targets with just a few data points. In comparison, a typical LIDAR point cloud data frame, sampled from surfaces in a scene can contain thousands or millions of data points. This is significantly higher than the data points collected from a scene using a mmWave radar. As we shall discuss later in this report, it is not possible to stream raw $I - Q$ data without additional hardware due to the memory and hardware limitations of the single-chip radar.

B. RADAR POINT CLOUD FORMATION

1) THE SIGNAL MODEL

Figure 3 shows an FMCW radar that transmits a series of chirps as shown in Figure 4. The transmitted chirp has the following characteristics; chirp duration T_r , sweep Bandwidth of the transmitted chirp B , amplitude A_{TX} and the initial

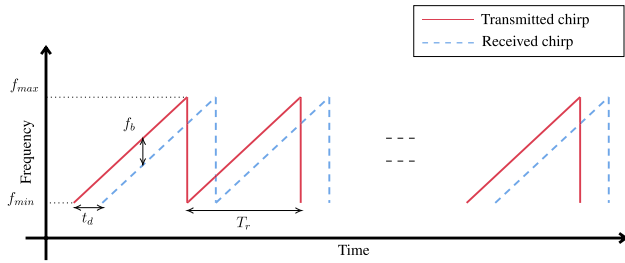


FIGURE 4. The transmitted chirp (—) and the received chirp signal (---). The synthesizer generates a sinusoidal waveform whose frequency is swept linearly from f_{min} to f_{max} . The duration of the chirp is T_r and t_d represents the time delay of the received signal. The beat frequency, denoted by f_b , is proportional to the distance of the target from the radar.

sweep frequency f_{min} . Mathematically, the FMCW transmit signal is given as [11]:

$$x_T(t) = A_{TX} \cos\left(2\pi f_{min}t + \pi \frac{B}{T_r}t^2\right) \quad (5)$$

From Equation 5, we can simplify $\phi_T(t) = 2\pi f_{min}t + \pi \frac{B}{T_r}t^2$. Assuming that we have a single target at a distance R from the radar, then the round trip delay $t_d = \frac{2R}{c}$ is given by $x_R(t) = \alpha x_T(t - t_d)$, where c is the speed of light and α is the path loss attenuation. In the radar processing chain shown in Figure 3, the transmitted signal x_T and the received signal x_R are mixed and the resulting Intermediate Frequency (IF) signal is given by $y(t) = x_R(t) \times x_T(t)$. This can be expressed further as:

$$y(t) = \alpha \cos(\phi_T(t - t_d)) \times \cos(\phi_T(t)) \quad (6)$$

$$y(t) = \frac{\alpha}{2} [\cos(\phi_T(t - t_d) - \phi_T(t)) + \cos(\phi_T(t - t_d) + \phi_T(t))] \quad (7)$$

and the expression $(\phi_T(t - t_d) - \phi_T(t))$, of Equation 7 can be calculated as:

$$\phi_0 - 2\pi \underbrace{\left(\frac{B}{T_c}t_d\right)}_{f_b} t \quad (8)$$

and this gives us the Beat Frequency f_b of the target.

2) RANGE, DOPPLER AND ANGLE ESTIMATION

a: RANGE ESTIMATION

According to Equation 8, the beat frequency f_b is the product of the frequency slope and the round trip delay of the signal, which is proportional to the distance of the object. Typically, an FFT (*range FFT*) of all the samples from a single sweep is used to convert the time domain IF signal into the frequency domain, where the peaks of the resulting Beat frequency spectrum give the range of the target.

b: DOPPLER ESTIMATION

From [11], an object in motion produces a Doppler frequency shift $\Delta\phi_f = \frac{2 \times v_r \times f_t}{c_0}$, where v_r is the radial velocity, f_t is the transmitted frequency and c_0 is the speed of light. In the context of radar systems, the radial velocity of the target is

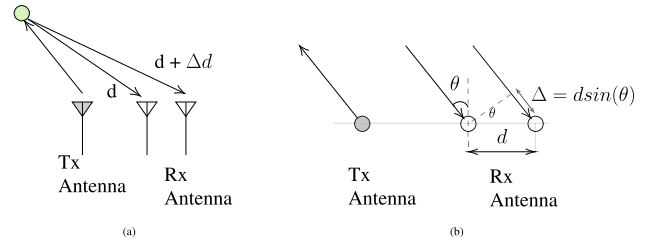


FIGURE 5. (a) one target, one Tx antenna and two Rx antenna configuration. Given that the Rx antenna separation is d , the target's radar signal will reach both antennas, but it will travel an additional distance Δd to reach the second antenna. (b) trigonometry diagram showing the determination of the angle of arrival (θ).

obtained from the phase change between consecutive chirps $\Delta\phi_v = 2\pi f_c \frac{2\Delta d}{c}$. Another FFT, (*Doppler FFT*) is then performed across chirps to estimate the phase change, and hence the radial velocity as shown in Equation 9. The resulting spectrum gives a 2D range-Doppler heatmap.

$$v_r = \frac{\Delta\phi_v \lambda}{4\pi T_c} \quad (9)$$

To produce a 2D signal $f(m, n)$ of size $M \times N$. Therefore, the 2D FFT to determine the Doppler is given by:

c: ANGLE ESTIMATION

One of the main requirements of a radar system is to estimate the angle of arrival (AoA) of targets in front of the radar. This is possible for radars that have multiple receiving antennas. Suppose there is only one Tx antenna and two Rx antennas, with a separation distance d as shown in Figure 5. The reflected signal from the target is received by both antennas, but the signal will travel an additional distance Δd to reach the second antenna. The additional time required to reach the second antenna is given by $\Delta t = \frac{d \sin(\theta)}{c}$. As mentioned earlier, a small change in the object position results in a phase change across the antennas given by $\omega = \frac{2\pi}{\lambda} d \sin(\theta)$ [11]. Assuming that the signal has a wavelength of λ , then the time difference between the signals arriving at the first antenna and subsequent antennas should be:

$$\Delta t = \left[0, \frac{d \sin(\theta)}{c}, 2 \frac{d \sin(\theta)}{c}, \dots, (N - 1) \frac{d \sin(\theta)}{c}\right] \quad (10)$$

Consequently, the phase difference between the signals arriving at the first antenna and subsequent antennas is given by:

$$\Delta\omega = \left[0, \frac{2\pi}{\lambda} d \sin(\theta), \frac{4\pi}{\lambda} d \sin(\theta), \dots, \frac{2\pi}{\lambda} (N - 1) d \sin(\theta)\right] \quad (11)$$

A third-dimensional FFT (*angle FFT*) is then used to estimate the phase change (ω) across the antennas. Finally, the AoA in Figure 5(b) can be derived from Equation 12.

$$\theta = \sin^{-1}\left(\frac{\omega \lambda}{2\pi d}\right) \quad (12)$$

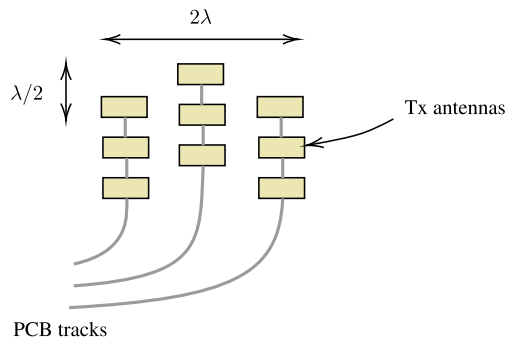


FIGURE 6. Transmit antenna arrangement for simultaneous azimuth and elevation angle estimation.

d: SIMULTANEOUS AZIMUTH AND ELEVATION ANGLE ESTIMATION

The antenna array in Figure 5 can estimate the azimuth angle if placed horizontally, and when it is oriented vertically, it can estimate the elevation angle. Simultaneous azimuth and elevation angle estimation is possible if the transmit antenna array is arranged, as shown in Figure 6. Basic principles of AoA estimation discussed in Section IV-B2.c apply here as well.

3) CFAR

The radar signal $x(t)$ received from the environment can be expressed as [11]:

$$x(t) = s(t) + n(t), \tag{13}$$

where $s(t)$ is the target reflection and $n(t)$ is white Gaussian noise. The purpose of a detector is to determine the presence or absence of a target in the received signal. Classical detectors such as the Neyman-Pearson detector use a fixed threshold factor. The assumption is that the interference is identically distributed over the test cells, such that if the signal in the test cell exceeds a certain threshold, then a target exists in that cell. This assertion can be misleading in certain circumstances, for example, when the target returns only contain interference (noise or clutter) that exceeds the detection threshold. This gives rise to False Alarm conditions as shown in Equation 14.

$$s(t) + n(t) \geq Pthr, \text{ True Detection} \\ n(t) \geq Pthr, \text{ False Alarm} \tag{14}$$

In order to address the occurrences of false alarms and missed detections caused by a fixed threshold, Constant False Alarm Rate (CFAR) detectors that rely on maintaining a constant false alarm rate, hence varying the detection threshold within cells have been developed. In this method, the detector estimates the noise level within a sliding window and uses this estimate to determine the presence or absence of a target in the test cell. If a target is present in a cell/bin, the algorithm returns the target’s localization in the range-Doppler domain.

Finally, all the CFAR detected targets are grouped based on their location on a 2D grid. Usually, a 3×3 kernel slides along

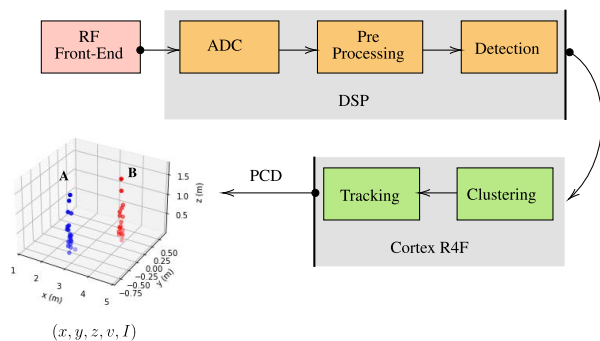


FIGURE 7. Flowchart of PCD generation. The radar processing chain consists of basic building blocks, such as an analog-to-digital converter, pre-processing block (multi-dimensional FFT operations), CFAR detection, clustering, tracking and finally a stream of the detected objects (A and B) on a 3D Cartesian plane. Each detected object is made up of individual points that have (x, y, z) information, as well as Doppler and the received signal strength.

TABLE 2. detObj data structure.

Number of objects	N
Range Index	r_i
Range	$\{1 \dots N\}$
Doppler Index	d_i
Doppler	(m/s)
Peak Val	dB
x	(m)
y	(m)
z	(m)

the detection matrix to compare the amplitudes of all the detected points within that kernel. With this approach, only the point with the highest amplitude is returned, while low amplitude points are discarded. This process is repeated until the entire detection matrix is exhausted. Thereafter, a third FFT is performed to estimate the target’s azimuth angle.

The result is an array of detected points (**Point Cloud Data**) $P = P_i(x, y, z)$, where $i = 0, 1, \dots, N - 1$, as shown in Figure 7 and each point p_i is associated with attributes such as: radial distance, cross-range, height, velocity and intensity.

4) mmWave POINT CLOUD DATA STREAM

The PCD is streamed from the radar front-end in a *detObj* data structure which stores the variables of the detected object(s) for each frame as shown in Table 2. From the data structure, we can extract five variables that are relevant for visualization, and consequently, feature extraction; $\{Doppler(m/s), peakval(dB), x(m), y(m), z(m)\}$. An observation of these points over consecutive frames gives more information about the nature or state of the target. Although the parameter *peak val* is barely used for feature extraction in its basic form, it plays a big role when the received data points are transformed into other representations, and the intensity information is added to highlight strong or weak reflectors on the target.

C. COMPARISON WITH OTHER RADAR DATA FORMATS

2D or 3D point clouds are different from 2D images or 3D data representation formats in several ways. Firstly, point clouds are permutation-invariant. This means that the data contained in $P = x_1, x_2, x_3, \dots, x_N$ remains unchanged for any permutation $\pi : \{1, 2, 3, \dots, N\} \rightarrow \{1, 2, 3, \dots, N\}$ of the individual points i.e $f(x, y) = f(y, x) \forall x$ and y . In other words, given a point cloud with N data points, there are $N!$ ways of representing this point cloud data as shown in Equation 15.

$$\bar{P} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} \quad \bar{P} = \begin{bmatrix} x_2 \\ x_1 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} \quad \bar{P} = \begin{bmatrix} x_2 \\ x_1 \\ x_4 \\ x_3 \\ x_5 \end{bmatrix} \quad \dots \quad \bar{P} = \begin{bmatrix} x_5 \\ x_4 \\ x_3 \\ x_2 \\ x_1 \end{bmatrix} \quad (15)$$

Functions that exhibit such features are called Symmetric Functions, and examples include: $\text{sum}(a, b)$, $\text{max}(a, b)$, $\text{min}(a, b)$, and $\text{avg}(a, b)$.

Another property of mmWave radar point clouds is irregularity. In basic point processing tasks, it is common to sample the points uniformly across the object. However, in the context of mmWave radar point cloud applications, it can be challenging to obtain points that are evenly spread across the target. In most circumstances, the number of points changes from frame to frame due to external factors such as noise, clutter, and multipath signals. Furthermore, the location of the detected peaks changes between successive frames, even for a stationary target, due to the variabilities of the peak grouping algorithm. This results in a non-uniform inter-frame point cloud distribution.

Lastly, the point cloud does not have regular grid-like 2D images or 3D volumes, as shown in Figure 8. As discussed earlier in this article, the points lie in 3D space where the distance to the neighbouring point is not defined, unlike images with equal spacing between pixels or volume cells. Additionally, a single point in space does not convey much information. It lacks the basic parameters such as length, dimension or area. It will take the effort of a sequence of these points to make sense of the information conveyed by the target.

Despite these characteristics, the radar point clouds provide robust and versatile geometric features that can solve some target detection, recognition and classification problems. According to the collected literature, many techniques have been developed to work with the point cloud data from mmWave radar devices. In particular, two methods exist:

- 1) Raw point cloud processing - Developing machine learning and deep learning models which consume the point clouds directly. Currently, there are four main classes of deep learning models for directly processing the features of static point cloud data. These are PointNets, specialized COVNETs, RNN-based and Autoencoders. For a dataset which contains point clouds of a moving target, PointRNN, a variant of PointNet has been developed. These models work well for the point

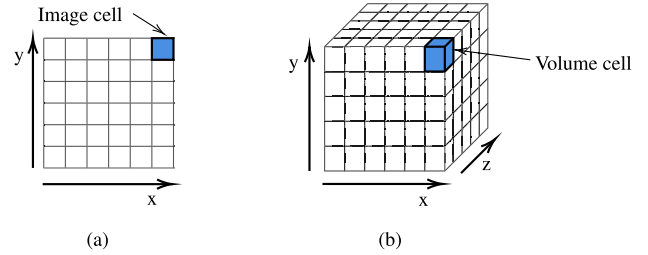


FIGURE 8. (a) an image cell (x_i, y_i) on a 2D grid and (b) a volume cell (x_i, y_i, z_i) on a 3D grid, where (x) , (y) and (z) dimensions have identical distances.

cloud dataset because they have building blocks which learn the local features of each point independently and then extract shape features with pooling layers.

- 2) Point cloud transformation - Developing methods which transform the point clouds to different information domains before being passed on for further processing. In the literature, most authors have converted the raw point clouds to:
 - Radar volume grid maps
 - Occupancy maps
 - 3D voxels
 - 2D Image tensors:
 - Range map,
 - Range-Doppler map,
 - Range-Doppler Azimuth map,
 - Micro-Doppler signatures.

As reviewed in the previous subsection, the raw point cloud data format is inherently different from conventional radar data formats, thanks to their unique properties. Nonetheless, several studies in the literature have provided means to either directly work on the raw point clouds or convert them to a form suitable for standard mathematical operations. The methods adopted for point cloud processing include PointNet, and its variants, while point cloud transformation creates range-time, range-azimuth, range-Doppler images, and 3D voxels. However, it must be noted that selecting a potential method for any application largely depends on the resources available and the constraints of the project. For example, deep learning on 3D voxels intuitively requires substantial computational and memory resources compared to consuming point clouds directly. In the future, more experiments are needed to ascertain this hypothesis.

D. mmWave RADAR POINT CLOUD PROCESSING

The received point cloud data cannot be used in its raw form. Other processes need to be performed first to guarantee the accuracy of the radar application. Thus, this section reviews the processing methods performed on the radar point clouds, particularly focusing on noise removal and feature extraction. First, noise removal through range gating is discussed, including clustering techniques in Section IV-D1.a and IV-D1.b respectively. Second, we elaborate on extracting geometry and statistical features in Section IV-D2.

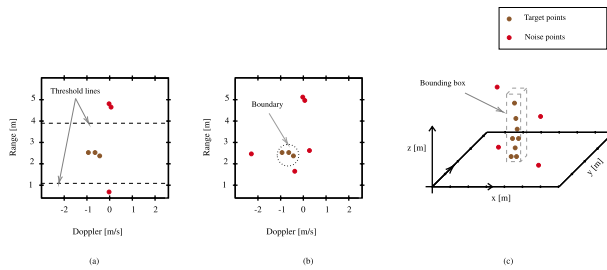


FIGURE 9. (a) noise removal through range gating. This process involves setting lower and upper range limits. Any points that do not lie within this region are assumed to be noise points. (b) Noise removal through 2D and (c) 3D point cloud clustering. This method creates clusters based on user-defined parameters such as the minimum number of points, distance to the nearest neighbor, the radius of the circle, or the height of the box. Any points that are not in any cluster are discarded, while points that are in a valid cluster are assumed to be reflected from the target.

1) NOISE REMOVAL

One of the challenges of using mmWave radar point clouds for target classification or recognition is that the output point cloud data is easily affected by environmental factors such as clutter and multipath reflections. The latter produces multiple data points that appear at random locations between successive frames of the radar image. As a result, it becomes difficult to model the Spatio-temporal characteristics of the target. In such circumstances, *noise removal* becomes helpful. It involves removing the data points that are not related to the target or region of interest.

Generally, there are two approaches to address this problem, (i) thresholding (range-gating), as shown in Figures 9a; or (ii) clustering, as shown in Figure 9b and 9c. These methods allow isolation of the points from the target while discarding the *unwanted points*. We illustrate these approaches in the following sub-sections. Finally, Table 3 provides a brief summary of these techniques.

a: RANGE GATING

Noise removal through range gating involves setting lower x_1 and upper range thresholds x_2 , such that $x_1(m) < R < x_2(m)$. This means that any data points that are not in the region R are treated as outliers, and are thus removed, while the data points within this region are assumed to be from the target, and are thus retained for further processing.

Noise removal through range gating in mmWave radars was initially mentioned by Adeel Ahmad et al [55] in 2018, where the authors developed a method of monitoring the vital signs of multiple people using an IWR1443 77-81 GHz mmWave radar. In their framework, the authors use range-gating to isolate the objects in the range-azimuth domain, thus separating the desired target from clutter. However, range gating requires knowledge of the target position beforehand, such that we get an estimate of the range thresholds. This means that such techniques may not be feasible for real-time automatic operations where the components of a target change range bins in consecutive frames. Therefore, potential application areas of range gating are thus limited to scenarios where the target or its components are at known distances from the radar, such as hand gesture recognition.

TABLE 3. Summation of noise removal techniques from radar point clouds.

State-of-the-art	Challenges	Opportunities
Range gating	Requires prior knowledge of the target position	Adaptive range thresholding
Clustering	Easily affected by data points from clutter.	1. Powerful density-based clustering algorithms, 2. Moving target indication, 3. Radar-Camera sensor fusion

b: CLUSTERING

Point cloud clustering is often considered an improvement to noise removal through range gating, as shown in Figures 9b and 9c. Clustering techniques group the data points based on two parameters, (i) radius of the neighbourhood and (ii) the minimum number of points in a neighbourhood [8]. In the case of a 2D domain such as range-Doppler or range-azimuth image, the clustering algorithm creates a circle of radius eps , and any points that exceed a certain minimum threshold $minPts$ are assumed to belong to the same cluster. Points that do not belong to any cluster are thus removed from the point cloud. 3D clustering requires the width (m), breadth (m) and height (m) of the bounding box, as shown in Figure 9c. However, a disadvantage of this approach is that clutter can also produce data points that meet the minimum requirements of forming a cluster, making it difficult for algorithms to distinguish between real clusters and those made from clutter.

A simple workaround to this problem is density-based clustering. Such unsupervised learning techniques promise great improvements in separating target clusters from noise [56]. Feng et al incorporated this concept in their framework to identify the scattering points of multiple patients in a room [56]. Next, they used a Kalman filter to track the movement of each cluster based on the assigned trackID [56]. In their work, the authors demonstrate the feasibility of extracting multiple clusters and monitoring the behaviour of each cluster over time. This is important in case there is a need to discard clusters that do not conform to the expected behaviour of the targets.

Beyond noise removal through thresholding, there also exist several techniques that address this problem. For example, Guangcheng et al demonstrate the concept of a moving target indication (MTI) in the context of an AWR mmWave radar. In their work, the authors use an MTI to separate the point cloud of human targets from clutter [57]. Typically, clutter sources are characterized by multipath reflections due to several objects in the scene, and this results in noise points appearing randomly in the point cloud [57].

2) FEATURE EXTRACTION FROM 3D POINT CLOUD DATA

Extracting features from mmWave point cloud data is one of the steps in automatic radar applications. This research area has received substantial attention since the radar was

introduced around 2015. In the literature, a large number of papers focus on developing innovative approaches to machine learning, focusing on novel application areas such as human gait recognition and automotive radar applications, among others. In this section, we discuss several techniques for extracting features from the mmWave radar point cloud data. We extend the study by discussing the advantages and disadvantages of using such features. We conclude the section by summarizing these techniques in Table 4.

Generally, two types of features can be extracted from the radar point cloud data. The first provides information about the shape or geometry of the point cloud distribution, while the second provides statistical information that describes the point cloud distribution.

a: GEOMETRY FEATURES

In many automotive radar application areas we have reviewed, (for example, human, cyclist, car and truck classification), the classes under study have distinguishable geometries. It is, therefore, a natural choice to use geometry information for such classification problems.

Considering Figure 10, it is clear that the point cloud distribution roughly approximates the target dimensions. That is to say, it is possible to draw an estimate of the target dimensions/area/volume from the plot. Therefore, we can extract the geometry of the target from the plot, and pass it on as a feature vector to the classification algorithms.

From this plot, we can construct a 2D bounding box, (X, Y) or 3D box (X, Y, Z) that describes the shape of the target. Consequently, the following features can then be defined from this bounding box [58]:

- 1) k_1 : the length of the bounding box, $(\max(X) - \min(X))$,
- 2) k_2 : the width of the bounding box, $(\max(Y) - \min(Y))$,
- 3) k_3 : the height of the bounding box, $(\max(Z) - \min(Z))$,
- 4) k_4 : the area of the bounding box, $(Length \times Width)$,
- 5) k_5 : the volume of the bounding box $(Length \times Width \times Height)$
- 6) k_6 : point cloud density: Number of point clouds in the bounding box $N/Area$

b: STATISTICAL FEATURES

Apart from the basic target dimension features discussed in Section IV-D2.a, a simple yet efficient statistical analysis of the point cloud distribution can yield new knowledge about the target. Here, the overall objective is to use the standard statistical distributions to analyze the characteristics of the target. We illustrate some of these features in the following paragraphs:

i) ARITHMETIC MEAN

Given a point cloud data stream $P = \{p_1, p_2, \dots, p_N\}$, we can extract the first dimension $p_1 = \{x_1, x_2, x_3 \dots, x_m\}$, where m is the total number of points in the data stream. The same applies for the different dimension p_2, p_3 up to p_N . Given that the total number of data points m , reflected from the target fluctuates from frame to frame, we can obtain the average

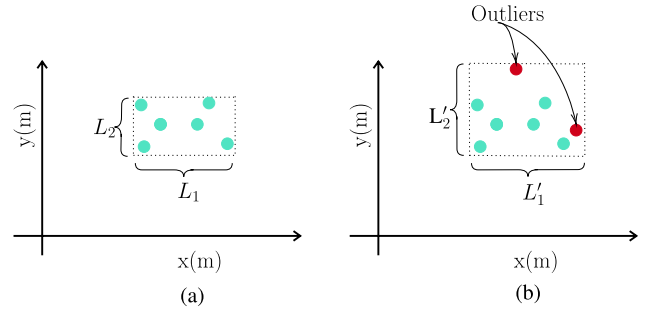


FIGURE 10. An illustration of the effect of outliers on the estimation of shape descriptors. L_1 and L_2 are features which describe the shape of the target on a 2-dimensional plane. The features are calculated from the received scattering points of the target and they are used to estimate the width and length of the target. In (a), we can extract these features with some approximation. However, in (b), the presence of the scattering points from clutter gives rise to random outliers which highly affect the calculation of the geometric descriptors (L'_1 and L'_2). This can be eliminated by taking a large sample size, however, clutter is random and the scattering points from noise or clutter can give inaccurate estimates of the size of the target, which may lead to classifier under-performance.

number of points received as:

$$\mu = \frac{1}{m} \sum_{i=1}^m x_i \quad (16)$$

ii) STANDARD DEVIATION

The standard deviation of the points of the different dimensions in the data stream can be defined as:

$$\sigma = \sqrt{\frac{1}{m} \sum_{i=1}^m (x_i - \mu)^2} \quad (17)$$

iii) CENTROIDS

The centroid of the PCD can be defined as an array of the averages of each dimension of the target's bounding box. Just like any data point, we have discussed so far, the centroid too is a point in 3D space. Referring to Figure 9c, where the point cloud of a person is defined by the bounding box $P = \{x, y, z\}$, and Equation 16, the centroid can be defined as $c = \{\mu_x, \mu_y, \mu_z\}$. The result is a data point c , that can yield new perspectives about the position or state of the target. For example, studying the temporal position of the centroid point relative to the xy plane (ground) can reveal the motion or orientation of the person. Potential application areas include fall detection and posture estimation.

Recent works confirm that geometric and statistical features extracted from the radar point clouds can be useful for target classification or fall detection for the elderly. For example, Zihao et al, developed a point cloud features-based kernel SVM for human-vehicle classification using a mmWave radar [58]. Powered by a kernel SVM classifier, the authors developed a human-vehicle classification system that uses eleven geometrical and statistical features extracted from the point clouds. Their results show superior classification accuracy compared to signal features.

In another study published recently, Feng et al developed mmFall, a framework that uses mmWave radar point clouds

TABLE 4. Summation of feature extraction from 3D point cloud data.

State-of-the-art	Challenges	Opportunities
Geometry Features	Inter-frame variability	Taking a large sample size.
Statistical Features	Easily affected by scattering points from clutter.	<ul style="list-style-type: none"> • Robust noise removal algorithms, • Sensor fusion,

and a hybrid variational RNN auto-encoder to detect when elderly people fall [59]. In their work, the authors combine the body centroid and the anomaly of the point cloud distribution to detect a fall. Their system registers a fall when the body centroid and the point cloud anomaly spike simultaneously [58]. Their results indicate that these features are suitable for modelling human fall motion, and have lower calculation costs than voxel processing. Thus, the authors were able to achieve impressive performances of 98% detection accuracy.

More recently, centroids were used by Peter et al in their pedestrian detection and tracking algorithm [60]. Specifically, their algorithm considers the position of the centroid points between successive frames to estimate the trajectory of the target.

Exploiting geometry and statistical features is simple, effective and works best at distinguishing classes with a large intraclass and low interclass variability. However, such features are vulnerable to noise points reflected from clutter, as demonstrated in Figure 10b. A stray data point alters the target dimensions (L'_1, L'_2), and hence yields incorrect geometry and statistical features. At present, a reliable workaround to this problem is to develop robust noise removal algorithms that efficiently remove the noise points due to clutter. Another approach that has been suggested to deal with multipath reflections is to conduct the radar detection experiment in an environment that does not have any other reflecting targets. However, in a typical radar deployment, it is difficult to completely eliminate such targets. Therefore, more work needs to be done to enable the use of statistical and geometric features in noisy radar environments.

A promising workaround is noise and outlier removal through sensor fusion, machine learning and deep learning methods. In essence, sensor fusion holds promise in this regard due to the use of inputs from disparate sources. The technology has already been tested in different application domains, however, its application in point cloud noise removal is promising, and is yet to be fully explored. Thus, noise removal in mmWave point clouds using sensor fusion and other techniques should be investigated in the future.

E. POINT CLOUD TRANSFORMATION

Point cloud transformation refers to the process of converting the 3D point clouds to a data representation format that

has a regular grid while maintaining the target information. Based on the data contained in the point cloud, 3D point cloud transformation can be classified into two categories: (1) construction of an occupancy grid and (2) construction of radar image tensors. For the first category, point cloud transformation takes the x, y, z coordinates of the received points and creates 2D or 3D maps with evenly spaced grids that represent the presence of a target. For the second category, point cloud transformation takes the range, azimuth, Doppler and time information to create 2D image tensors that are similar to images obtained from FFT operations. The two categories are summarized in Sections IV-E1 and IV-E2 respectively, as well as in Table 6.

1) POINT CLOUDS TO VOXELS

A central feature of 3D voxels (volume elements or volume pixels) is that they have a well-defined 3D grid that is uniformly spaced along each dimension, making it possible to apply the standard deep learning convolution operations that have been developed for volumetric images [26]. Unlike raw point clouds, voxels provide a well-structured volumetric representation of the target, enabling the extraction of geometrical features such as target dimensions, the volume occupied by the voxels, or point density within the voxels [61]. It is also possible to extract Eigen-based features such as omnivariance, surface variation, and linearity, among others [61]. In the literature, voxel-based representation has shown great potential in areas such as activity recognition, human tracking, and pose estimation among others. A brief overview of such applications is shown in Table 5.

Voxelization converts the sparse point cloud data into a discrete representation of the detected targets on a 3D voxel grid. This algorithm creates a volume cell that best approximates the structure of the target based on a point's (x, y, z) coordinates. Taking a discrete 3D voxel cell as shown in Figure 11b, a voxel that is created from a discrete point at coordinates (x, y, z) is defined as a continuous region of space (j, k, l) such that: [69]

$$\begin{aligned}
 x - 0.5 < j \leq x + 0.5, \\
 y - 0.5 < k \leq y + 0.5 \text{ and} \\
 z - 0.5 < l \leq z + 0.5.
 \end{aligned}
 \tag{18}$$

Thus, each voxel $V_{j,k,l}$ is a unit volume cell or cube that is centered at (x, y, z) and has the following properties:

- 1) Six faces,
- 2) Twelve edges and
- 3) Eight corners.

Generally, there are five steps involved in the transformation of point clouds to voxels, as shown in Figure 11c-d. The first step is the computation of the initial 3D bounding box. In essence, this step calculates the length, width and height (X, Y, Z) of a large cuboid within which all regular voxels lie. In the second step, the 3D bounding box is subdivided into a subset of voxels whose dimensions are given by $\Delta x, \Delta y, \Delta z$ respectively. Afterwards, the number of voxels

TABLE 5. Summation of Voxel research.

Ref	Work	Notes
Arif <i>et al</i> [62], 2021	A multiple-inhabitant activity recognition system	Applied a voxel fitting algorithm on PCD
Peixian <i>et al</i> [63], 2021	To identify boxing, jack, jump, squats, and walking.	SVM voxels produce the least accuracy of 63.74% while a Time Distributed CNN + Bi-Directional CNN produces the best accuracy on voxels of 90.47%. However, Graph based methods produce the best accuracy and are promising for point cloud data.
Ziheng <i>et al</i> [64], 2020	To track the motion of the thumb in real-time and move the cursor on a computer screen.	create a $25 \times 25 \times 25$ voxel
Akash <i>et al</i> [26], 2019	Human activity recognition, walking, jumping, jumping jacks, boxing.	Created voxels of dimensions $10 \times 32 \times 32$ Choice of dimension reached by testing the performance.
Salami <i>et al</i> [65], 2021	Developed a gesture recognition system Compared a Graph convolution approach with SVM, MLP, Bi-Directional LSTM RadHar that use voxels as input.	Their model outperforms the baseline standards in Accuracy and AUC.
Arindam <i>et al</i> [10], 2021	Leverage developments made in Natural Language processing to estimate the skeletal pose using radar point clouds. They manage to estimate up to 25 skeletal key-points.	Firstly, the radar point clouds are voxelized and then passed on the mmPose-NLP architecture where the voxel indices of the skeletal key points are predicted and then the indices are converted back to 3D coordinates using the same dictionary.
Andriam <i>et al</i> [66], 2020	Skeletal pose estimation using NLP techniques.	Simulate radar point clouds using Kinect skeletal tracker and a MATLAB API.
Yue <i>et al</i> [67], 2021	Develop a system to Image and Reconstruct 3D shapes of objects using mmWave radar point clouds.	They first create voxels to group the points.
Yuheng <i>et al</i> [68], 2021	Develop a human activity recognition system to detect five activities: "Boxing, Jumping, Jumpjack, Squats and walking". Different from RadHar in terms of practicality.	RadHar is based on offline processing. They voxelize the points from human radar returns, and they create 60 voxels of dimensions $(30 \times 15 \times 20)$

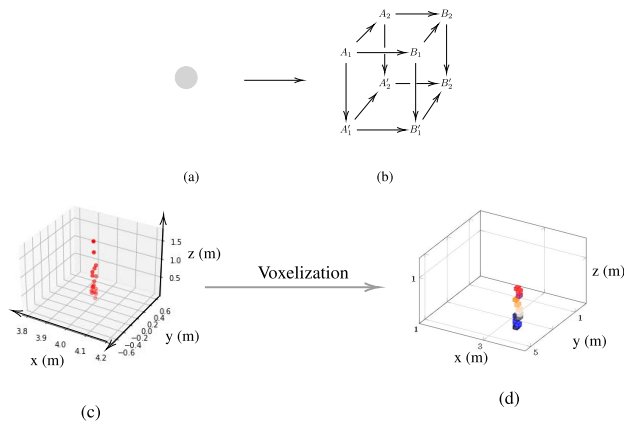


FIGURE 11. Converting a data point (a) to a 3D voxel cell (b). Point clouds (c) of the form $\{P_i | i = 1, 2, 3, \dots, n\}$, where P_i has the coordinates $x(m)$, $y(m)$, $z(m)$ are converted to a voxel image (d) with dimensions of $J \times K \times L$. For example, 50 voxel images will have the dimensions of $(50 \times 25 \times 25 \times 25)$.

(N_x, N_y, N_z) along each dimension is computed, along with the voxel indices (i_x, i_y, i_z) . Finally, the voxels are calculated. Algorithm 1 summarizes these steps.

The concept of voxelizing point clouds is not entirely new. In fact, this method has been studied widely in LIDAR point cloud processing. Even though LIDAR sensors and mmWave radars are fundamentally different in operation, it makes sense to transfer the knowledge gained in LIDAR point cloud processing to mmWave radar point cloud processing since they all use point clouds. One such area is the use of deep learning methods for voxel classification. This

Algorithm 1 Voxelization Algorithm

- 1: **procedure** Voxelize_Point_Cloud(x, y, z)
- 2: Create an initial 3D bounding box,
- 3: Subdivide the bounding box into a subset of voxels,
- 4: Determine the size of the voxels $(\Delta x, \Delta y, \Delta z)$,
- 5: Calculate the number of voxels along each dimension:

$$N_x = \frac{(x_{max} - x_{min})}{\Delta x} + 1,$$

$$N_y = \frac{(y_{max} - y_{min})}{\Delta y} + 1,$$

$$N_z = \frac{(z_{max} - z_{min})}{\Delta z} + 1$$

- 6: Compute the voxel indices:

$$i_x = \frac{(x - x_{min})}{\Delta x}$$

$$i_y = \frac{(y - y_{min})}{\Delta y}$$

$$i_z = \frac{(z - z_{min})}{\Delta z}$$

- 7: Check if all voxels were created

$$i = i_x + i_y \times N_x + i_z \times N_x \times N_z.$$

- 8: **Return** Voxels

field has experienced tremendous growth over the past years and a number of works have sought to determine standard deep learning models specifically for voxel classification.

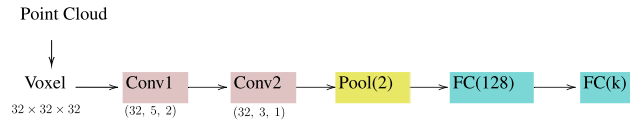


FIGURE 12. Architecture of the voxNet framework for 3D real-time object recognition [70]. From left to right, point clouds coming from the depth sensor are converted to $32 \times 32 \times 32$ voxels. The resulting occupancy grid is processed by two convolutional layers, which extract features from the voxels. A pooling layer downsamples the input by a factor of two. Finally, two fully connected layers form the last layers of the voxNet, with a softmax nonlinearity to provide the classification result.

The voxNet framework [70], shown in Figure 12 is one such example.

This framework has been designed to accept real-time point cloud data, convert them to a fixed size 3D occupancy grid, and then a series of convolutional, pooling and fully connected layers work in unison to give the predictions. Another similar framework that accepts voxels is the Point-Voxel CNN (PVCNN) [71]. The framework was developed to reduce the memory consumption of point-based input data, while leveraging voxel convolutions to improve the locality [71]. The result is a high-performance framework that is two times faster than models that have been primarily developed to consume point clouds. These models have demonstrated success in LIDAR point cloud processing, however, there has been no detailed investigation on the use of such baseline models to classify the sparse mmWave point cloud radar data. Nevertheless, the authors explore different deep learning architectures for processing voxels created from mmWave devices, and they have reported good performances as highlighted in Table 5.

On the other hand, voxelised representations of point clouds have their unique disadvantages. Mainly, the accuracy is highly affected by the resolution of the voxel grid. Taking the sparse point clouds from the mmWave radar, for instance, a larger resolution covers the entire point cloud but overshadows finer differences, meaning that spatial information is lost. At the same time, finer voxel resolutions have a tendency of leaving empty voxels. This problem is apparent in tasks with low interclass variations, such as sign language or hand gesture recognition. Such tasks would require the use of low voxel resolutions to reduce spatial information loss and hence identify the different gestures. However, the use of finer resolutions further translates to increased memory consumption and places heavy computational workloads on the processor.

The second major challenge relates to computational efficiency and memory footprint. For example, a $312 \times 312 \times 312$ volumetric display may produce up to 30 million voxels, that have to be passed on to the deep learning models for feature extraction. Lastly, the algorithm to convert the point cloud data to voxels (*points_to_voxels*) requires a significant amount of time to compute the voxels, hence, voxel-based data representation may not be suitable for real-time operations. Nevertheless, these problems can be mitigated by

storing the voxels in memory using binary tree structures [61] and octrees [69].

In summary, voxels provide a convenient way of transforming the point cloud data into a form that has a regular grid. Afterwards, it becomes easier to extract a couple of user-defined features or leverage the standard deep CNN frameworks that have been developed for volumetric problems. Nevertheless, the new 3D structure comes with its own set of challenges, mainly, the voxel resolution. Improper selection of the resolution greatly affects the overall performance. Therefore, a lot of factors must be considered first before adopting the voxel structure. In principle, voxels work best in classifying classes with a large variance such as humans and vehicles, as compared to tasks with a low variance such as sign language recognition. Further research should be undertaken to improve the performance of deep learning algorithms on voxel-based data representation.

2) POINTS TO 2D IMAGES

Radar images are typically obtained by performing a multidimensional FFT on the ADC samples. The result is a sequence of 2D images that conform to the requirements of the deep learning models. Several types of images can be created this way. Examples include a range-time map, range-Doppler map, Doppler-time map and range-azimuth map. The most popular is the micro-Doppler plot, thanks to its ability to display the periodic movement of components on the target. In this section, we review some of the common approaches to creating micro-Doppler images from point cloud data or range-Doppler frames.

As discussed in Section IV-B2, a range-time map shows how the range of a target varies over time, while the range-Doppler map shows the range of the targets and their corresponding Doppler characteristics. In other words, stationary targets will appear at their respective distances while occupying the zero Doppler bins, and objects in motion will show a nonzero response in the Doppler spectrum. A sequence of range-Doppler maps shows the behaviour of the target over time and also infers the nature of the radar target when it is composed of multiple scatterers.

It is also possible to obtain 2D radar images (2D heatmaps) directly from the 3D point cloud data. By projecting the multi-dimensional point clouds onto the various domains, as shown in Figure 15, researchers could obtain 2D images that carry enough information just like regular images obtained through conventional FFT operations. In this way, 2D image tensors such as range-time and range-Doppler maps can be obtained from a sequence of range-Doppler frames. But first, the projected images still lack a standard 2D grid and connectivity information, as shown in Figures 15 (b) and (c). Therefore, a second mathematical operation is required to convert the projected images into a structure suitable for performing 2D convolutions.

However, the images obtained this way do not have the same resolution as the images produced by FFT operations. Nevertheless, the information they display is sufficient for

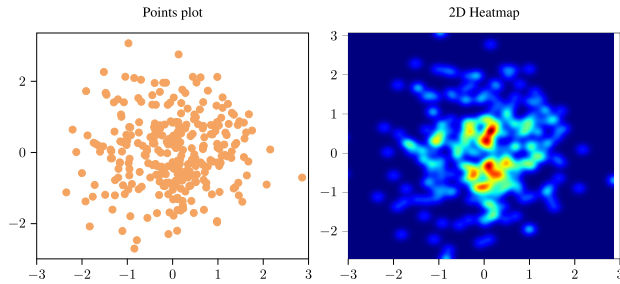


FIGURE 13. Points to Image conversion using the workflow outlined in Algorithm 2. This function takes 2D points that do not have connectivity information and creates a heatmap image that has a 2D grid. Additional parameters such as smoothing may be added to produce an image with good resolution.

target detection, recognition and classification tasks, as we shall prove later in this section. Generally, the challenging task concerning mmWave point cloud projection is to develop a *points_to_image* algorithm that preserves the target's information and also avoids a lot of empty pixels in the final 2D image due to sparsity. Although empty pixels do not pose much of a problem at this stage, a simple workaround to this problem is to insert some data points into the raw point cloud data [72], to produce a continuous image. However, it should be noted that the application area must be considered first before adopting point cloud interpolation as a means of increasing the point cloud density.

A practical method to convert the point clouds to an image is relatively simple and straightforward. The method accepts x and y coordinates of the points, as well as the number of bins. These variables are passed on to a 2D histogram function that computes the heatmap of the points. This heatmap has a 2D grid, and is thus suitable for feature extraction and classification using standard machine learning and deep learning algorithms. Algorithm 2 summarizes the general workflow, and Figure 13 shows the expected output.

Algorithm 2 Point to Image Algorithm

- 1: **procedure** Points_To_Image(P)
 - 2: Point dataset: $P = \{x_1y_1, x_2y_2, \dots, x_Ny_N\}$
 - 3: $grid_size \leftarrow size\ of\ grid$
 - 4: $h \leftarrow height\ of\ grid$
 - 5: $x_{min} \leftarrow \min(x)$
 - 6: $x_{max} \leftarrow \max(x)$
 - 7: $y_{min} \leftarrow \min(y)$
 - 8: $y_{max} \leftarrow \max(y)$
 - 9: Construct grid
 - 10: Calculate intensity
 - 11: **Return** Heatmap
-

Similarly, Renyuan et al present their system for detecting the motion of a person in real-time using mmWave radar point clouds and a CNN [8]. In their framework, they pass a series of range-Doppler plots through their *Doppler – time* extraction module. The basic idea is to create a standard micro-Doppler plot from the range-Doppler sequence frames

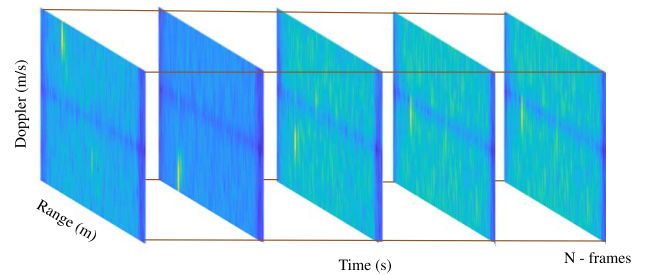


FIGURE 14. An illustration of consecutive range-Doppler heatmaps from a swinging pendulum. The stacked 2D range-Doppler heatmaps form a 3D Range-Doppler-Time matrix. From this representation, it becomes possible to extract other information domains such as range-time and Doppler-time heatmaps. This is a possible way of extracting the temporal variation of the range, or Doppler of targets using resource-constrained single-chip mmWave radars.

to extract the time-varying 2D features for target classification. The authors sort the received point cloud into a Doppler histogram with respect to time. Additional parameters such as RSSI are added to the respective Doppler bins to highlight their intensities. The result is a micro-Doppler plot that can distinguish the following human activities with high accuracy: a person walking towards the radar, a person waving hands, a person sitting, then walking and a person walking to and from the radar [8].

In the same way, Feng et al develop their mmWave radar point cloud classification and interference mitigation system [56]. They develop a framework that uses radar points clouds to create micro-Doppler plots that differentiate the following human activities; walking, falling, swinging hand for help, seizure and restless movement. Their Doppler pattern collection method utilizes a sliding window of one-second intervals to collect the Doppler bins of the scattering points with the same trackID [56]. In all, the produced micro-Doppler images are similar to the micro-Doppler images obtained through conventional FFT operations.

The second approach for creating 2D micro-Doppler images from the received data stream is to buffer the range-Doppler frames, as shown in Figure 14, and then extract the temporal variation of the Doppler. Although this method relies on the high-level range-Doppler information, this is a better approach compared to the previous method that relies on point cloud transformation. Firstly, the point cloud data only lists the detected objects, this means that some information that could be vital for target classification has been lost through CFAR peak detection. On the other hand, extracting the temporal variation from a sequence of range-Doppler heatmaps ensures that all the information obtained from the Doppler FFT will be used to create the micro-Doppler image. Secondly, micro-Doppler images obtained from point cloud to image conversion have a low resolution, unlike the images obtained from range-Doppler concatenation.

Recently, some authors created micro-Doppler images from a sequence of range-Doppler frames and then passed on the resulting plots to the deep CNN models for

TABLE 6. Summation of point cloud transformation techniques.

State-of-the-art	Challenges	Opportunities
Point clouds to voxels	1. Application area is determined by voxel resolution, 2. Higher voxel resolution decreases computational efficiency 3. Lower performance in tasks with low interclass variations.	Voxel-specific Deep learning algorithms.
Point clouds to 2D images	Poor image resolution due to point cloud sparsity	Produce point clouds with: <ul style="list-style-type: none"> • High range and Doppler resolution, • Maximum detection points per frame • Maximum number of frames per second (FPS)

activity classification. For instance, Kang et al used an FMCW radar to classify human activity by extracting features from micro-Doppler plots [73]. They extract Doppler profiles from each range-Doppler frame and concatenate the Doppler profiles along the range bins to construct a micro-Doppler image. The authors use an AWR1642 mmWave radar with the following specifications: Range resolution 4.74cm, Doppler resolution 0.07 m/s, 256 samples per chirps, and 250 chirps per frame form the data cube. These specifications produce a high-resolution micro-Doppler image that enables the extraction of motion statistics and motion pattern features with high accuracy.

In summary, the work by [8], [56], and [73] offer significant insights into the projection of mmWave radar point clouds to different domains. To obtain a high resolution or “continuous image” that is similar to images obtained through FFTs, the mmWave radar must be configured to (i), have a high range and Doppler resolution and (ii), produce maximum detection points per frame, and lastly (iii), the maximum number of frames per second (FPS). This is to reduce the occurrence of “empty pixels”, a phenomenon that may reduce the performance of the classification algorithms.

F. RADAR POINT CLOUD SIMULATION

It is well known that the performance of deep learning models is influenced by the size of the training dataset. A larger training dataset translates to improved model accuracy. However, it is difficult to obtain a relatively larger dataset on experiments alone. Furthermore, there may be a need to obtain the target’s responses at different orientation and aspect angles, something difficult to obtain in real life. Therefore, creating a synthetic dataset through simulations can help to increase

the size of the dataset. Unlike conventional data augmentation techniques such as cropping, flipping or resizing, radar point clouds present significant challenges due to their unique properties. In this section, we highlight possible techniques that can be used to simulate the radar point clouds of different targets.

Radar point cloud simulation entails answering the following two broad questions, (i) How can we generate synthetic mmWave radar point clouds for training the deep learning models? and (ii) How can we validate the generated mmWave radar point clouds?

Basically, the radar point clouds can be thought of as discrete points representing the dominant scattering areas of a complex target. Developments in computational electromagnetic modelling, such as the Shooting and Bouncing Ray (SBR) technique [74] and the Hough transform [75], enable the extraction of the target’s scattering centers from its 3D model under various scenarios.

1) SHOOTING AND BOUNCING RAY (SBR) TECHNIQUE

The SBR technique for analyzing the electromagnetic reflection of complex targets has been studied widely since the 1980s. In particular, one of the major application areas of this technique is in radar cross-section studies [76]. Here, the main objective is to determine the target’s scattering characteristics for model-based target recognition applications [76].

Another application area of the SBR technique is in Scattering Center Modelling. In its simplest form, the scattering center model provides a set of discrete scattering centers/points distributed throughout the target. This helps us to identify signatures, or points on the target that contribute to the overall *radar visibility* of the target, at certain azimuth and elevation (θ , ϕ) angles.

Ballah et al developed an automated technique to extract the 3D scattering centers from the target’s geometrical CAD model [77]. They developed a two-step framework that relies on the SBR technique. In the first step, the authors generate a 3D high-resolution image of the target based on the one-look ISAR algorithm. The image displays a heatmap, that has intensity information of the dominant scattering areas of the target. In step two, the authors extract the 3D position and strength (x , y , z , I) of the scattering centers from the ISAR image. Similarly, ray tracing has also been studied in [34], where the authors use the same technique to extract the scattering centers of a target from its respective 3D CAD model.

Their work demonstrates that ray tracing can be a useful tool for extracting the scattering points of targets. Such information can be used for signature compression and scattering center modelling, but most importantly, their work demonstrates how to generate a large dataset of the point cloud data for any 3D CAD model.

However, one key challenge of this approach is that it is only an approximation method [77]. This means that the extracted scattering points from the CAD models may be at slightly different locations than those extracted from

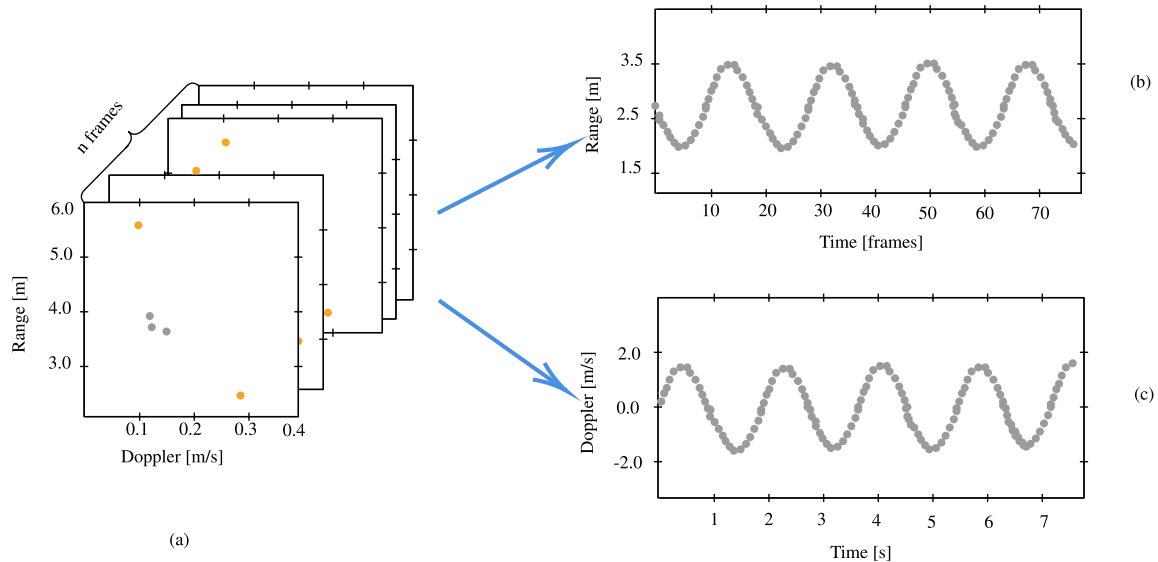


FIGURE 15. (a): scatter plot of the point cloud data of a target representing a single frame. The frames are stacked consecutively over the entire duration of the experiment to create a three-dimensional matrix. From here, different viewing frames can then be extracted so as to examine the target in the Range-time domain (b) or Doppler-time domain (c).

experiments. This leads to target recognition challenges, especially in mmWave systems where the resolutions are high. The workaround to this problem is to validate the simulation model through experiments and express the difference in the form of the Mean Squared Error.

While it is currently possible to simulate the scattering points from static 3D CAD models, a dynamic ray tracing technique can be a meaningful approach to simulate the point clouds of various movements, especially of a target that can be characterized by a periodic motion.

2) COMPUTER GRAPHICS TOOLS

Open source 3D computer graphics and modelling tools, such as Blender, have been shown to generate 3D point clouds from CAD models. However, a lot needs to be done to make them feasible for modelling radar systems.

G. DATASETS

In this section, we turn our discussion to publicly available datasets that have been gathered with mmWave radar sensors. The scope of the survey is only limited to datasets that contain raw point clouds or IQ data obtained from such devices. Our survey focuses on the source of the datasets, year of publication, data collection environment, potential application areas of the datasets, type of mmWave radar, and the subjects involved.

The MMAActivity Dataset, October 2019:

To facilitate accurate human activity recognition using mmWave radar, Akash Deep et al presented the MMAActivity dataset for human activity recognition using radar point clouds [26]. The dataset contains recordings of two subjects performing simple activities such as walking, jumping, jumping jacks, squats, and boxing. Each activity lasts for about 20s, creating a total of about 93 minutes of data. The IWR1443BOOST mmWave radar was used to collect the

point cloud data. The radar was placed 1.3m above the ground and the sampling rate was set to 30 frames per second.

From this dataset, the authors create a series of $10 \times 32 \times 32$ voxels and pass it on to the different classifiers. Various classifiers were evaluated and a combined Time-distributed CNN + Bi-directional LSTM classifier achieved an overall accuracy of 90.47%, compared with single models. This dataset is available on <https://github.com/nesl/RadHAR> and has been referenced by several authors, especially those who specialize in human activity recognition.

mmWaveRadarWalkingDataset, August 2020:

Ennio et al presented a raw IQ dataset of people performing various activities [78]. Overall, it contains recordings of 29 people of varying age groups, weights and heights performing 6 different activities in an indoor environment. During each recording period, the subjects perform the movements naturally (*i.e.*, hiding bottle, limping, slow walk, fast walk swinging hands), without following a pattern.

The data was collected using a TI IWR1642 mmWave radar sensor and a DCA1000EVM to provide the raw IQ data. This dataset is particularly useful for human gait studies using microDoppler analysis. With modifications to the radar signal processing chain, developers can obtain the point cloud data of human targets performing such activities. Its available in two parts: mmWaveRadarWalkingDataset_part1: <https://doi.org/10.5281/zenodo.3824534> and mmWaveRadarWalkingDataset_part2: <https://doi.org/10.5281/zenodo.3897234>.

CI4R- Human-Activity-Recognition-Datasets, May 2020:

The Laboratory of Computational Intelligence for Radar (CI4R) presented their human activity recognition dataset for microDoppler target classification [79]. Their dataset consists of raw I/Q samples of 6 subjects performing 11 different activities. Each subject performs 10 repetitions for each activity, creating a total of 60 samples for each activity.

The data was collected using 3 sensors for different frequency bands; 10GHz, 24GHz and 77GHz. A TI AWR1443 mmWave radar, and the DCA1000EVM provided the raw IQ data for the 77GHz frequency band. The data is available on <https://github.com/ci4r/CI4R-Activity-Recognition-datasets> and can be used to support multi-frequency gesture recognition algorithm development.

Multiple Patients Behavior Dataset, 2020:

The multiple patients' behavior dataset is an initiative by Feng et al to address the gaps identified in radar-based patient monitoring systems [29]. This dataset contains the point cloud data of one person performing five different activities (*i.e.*, walking, falling, swinging hand, seizure and restless movement). If no class is detected, the output class is "other behavior". Each recording lasts for about 30s and is made up of a Doppler pattern of the behavior. The dataset was collected using a TI AWR1642BOOST mmWave radar.

The multiple patients' behavior dataset is generally useful for training human activity recognition algorithms using radar point clouds. Potential application areas include patient monitoring, human tracking and elderly fall detection. The dataset is available on https://github.com/radar-lab/patient_monitoring.

mmFall Dataset, Mar 2020:

The mmFall point cloud dataset focuses on fall detection using a mmWave radar [59]. The dataset contains recordings of two subjects performing activities of daily living. This dataset is different from the datasets discussed previously in that the authors simulate a variety of fall patterns. They focus on random walking followed by a forward fall, backward fall, right fall, or left fall. The dataset was collected using a TI AWR1843BOOST mmWave radar and is available on <https://github.com/radar-lab/mmfall>. Potential application areas of the mmFall dataset include human tracking, patient monitoring and elderly fall detection.

multimodal traffic monitoring, 2020:

The multimodal traffic monitoring dataset presents point cloud recordings of a pedestrian and a vehicle collected in a car park [80]. The dataset consists of 9257 frames with a duration of 15 minutes. A TI AWR1843BOOST was used to collect the data. The data is useful for point cloud segmentation, vehicle and pedestrian tracking, and is publicly available on https://github.com/radar-lab/traffic_monitoring.

The Pantomime Dataset, March 2021:

The Pantomime point cloud dataset [81] comprises a set of 21 mid-air hand gestures collected from multiple indoor environments (*i.e.*, open environment, office environment, restaurant environment, factory environment and open environment with multiple people). The data was collected using a TI IWR1443 mmWave radar sensor. Overall, it consisted of data obtained from 41 participants of varying age groups, weights and heights. The gestures last 1–3 seconds on average, creating a total of 18.5 hours of recording time. During each recording, the participants perform easy hand movements and

complex hand movements at a distance of up to 5m from the radar. In total, the dataset contains about 22000 hand gesture instances.

The Pantomime dataset achieved a 95% accuracy and 99% AUC on a PointNet++ and LSTM hybrid architecture. This dataset belongs to the RadioSense project and is available on <https://doi.org/10.5281/zenodo.4459969>. This dataset has been used by other authors. Notably, it has been recently used by Salami et al [65], to develop Tesla-Rapture, which is a Graph-based CNN model to improve the performance of gesture recognition.

The MARS Dataset, October 2021:

Sizhe et al presented the MARS dataset comprising a synchronized mmWave radar and Kinect sensor to develop an assistive rehabilitation system for smart healthcare [82]. The dataset contains point cloud data generated by a TI IWR1443 mmWave radar. Overall, it contains data from 4 participants performing 10 different movements. They build a framework to reconstruct the 19 human joints from the 3D coordinates of the point cloud data.

The MARS dataset is available on <https://doi.org/10.1145/3477003> and is particularly useful for pose estimation studies.

The Open Radar Dataset, 2021:

The Open Radar Dataset is an initiative aimed to promote the sharing of radar datasets. Driven by the lack of open datasets for radar applications, Daniel et al provided their raw dataset of outdoor moving objects to assist algorithm development and benchmarking micro-Doppler target recognition algorithms [83]. Their dataset comprises raw data of 4 classes (*i.e.*, person walking, person cycling, UAV and vehicle).

The data was collected using a TI AWR2243BOOST mmWave radar, and the DCA1000EVM provided the raw IQ data. This dataset is particularly useful for developing target classification, detection, tracking and signal processing algorithms for moving targets such as people walking and cycling. Potential application areas include pedestrian and cyclist detection in automotive scenarios. The Open Radar Dataset is owned by the Open Radar Initiative and is publicly available on https://github.com/openradarinitiative/open_radar_datasets.

mmPose-NLP Dataset, 2021:

Sengupta et al presented mmPose-NLP: a dataset for skeletal pose estimation using a mmWave radar [10]. Their dataset contains 15000 frames of the point cloud data of two human subjects performing four different actions. The recordings focus on walking, left arm swing, right arm swing, and both arms swinging. The data was collected using a TI AWR1843 mmWave radar and it is available on <https://github.com/radar-lab/mmPose-NLP>.

1) LESSONS LEARNT

From this summary, it is clear that the majority of the published datasets target human gait recognition and human activity classification. Other application areas include human/cyclist detection and tracking in the context of

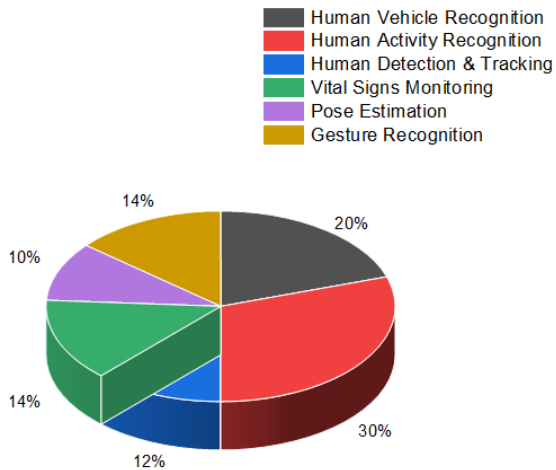


FIGURE 16. Number of collected papers for the application of mmWave radar point clouds. Several authors have applied mmWave radars in the field of human activity recognition, due to the increasing demand to detect falls, especially for patients, and the elderly. On the other hand, pose estimation using mmWave radars has received the least attention in the literature.

advanced driver assistance systems. Although datasets that target other areas such as drone detection are rare, several authors have worked in those fields nonetheless. While these open datasets encourage research in artificially intelligent systems, there is still a lack of benchmark datasets and metrics to evaluate machine learning or deep learning algorithms on radar point clouds. The Open Radar Initiative attempts to address this issue by providing its raw I/Q data of walking, and cycling people, UAVs and motor vehicles [83]. However, utilization of these datasets by different authors remains generally low in this field, mainly due to different experimental set-up scenarios and requirements.

V. CURRENT STATE OF THE ART

Single-chip mmWave radar products have been recently gaining popularity in new application areas because they have superior advantages over the majority of commercial radar products which are available on the market today. As a result, many authors have effortlessly developed novel applications which otherwise seemed impossible to do in the past. According to the literature cited, mmWave radar point clouds are applied in a wide range of applications including human vehicle recognition, human gait analysis, human detection & tracking, activity recognition, gesture recognition, patient & driver vital signs monitoring and pose estimation. This information is summarized in Figure 16. In some parts of the literature, there is no distinction in applications such as human gait analysis, human activity recognition and gesture recognition, making this category the highest application area of mmWave radars. In what follows, we review the current state-of-the-art application areas of mmWave radars, which we summarize in Table 7 & 8. We conclude the section by summarizing the trends and challenges that are associated with TI's mmWave radar products.

A. HUMAN VEHICLE RECOGNITION

Human vehicle recognition refers to the process of identifying pedestrians, cyclists, animals and vehicles from automotive radar spectra [98]. This area was once dominated by cameras and LIDAR sensors, but the introduction of all-weather, lightweight and low-power mmWave radar sensors offered several advantages over traditional sensors.

In 2018, Keegan et al demonstrate the feasibility of TI's IWR1642 mmWave radar sensor for traffic monitoring applications. In their work, they showcase a robust traffic and intersection monitoring system that uses mmWave sensors to detect moving vehicles [22]. Besides concept demonstration, they highlight (1) positions where the sensors can be positioned at the intersections and (2), the radar input parameters for medium-range MIMO applications and long-range MIMO vehicle detection. In their results, they report detecting moving vehicles up to a maximum distance of 150m and speeds in excess of 300 km/h.

Instead of just detecting moving vehicles at an intersection, Peter et al shed light on the optimization of the traffic flow by moving from a static traffic light control system to a dynamic system that is controlled automatically by the presence of pedestrians [60]. In their framework, a tracking algorithm is exploited to identify point cloud clusters of pedestrians at a crossing and trigger changes in the traffic light sequence. The authors use an IWR6843AOPEVM mmWave sensor and they demonstrate superior performance over camera-based systems.

In other studies, machine learning models have been used to study the spatial dependencies in the data of the different target classes. For example, Zhao et al, consider a Kernel SVM for human-vehicle classification using eleven features extracted from the point cloud distribution [58]. In another study, Jin et al consider a GMM for point cloud segmentation in the context of traffic monitoring [80]. However, most studies adopt deep learning models to classify the radar returns [87]. In most cases, the authors create a microDoppler dataset of the different targets and use it to train the deep learning models. As we can expect, such frameworks offer great improvements in performance, as already shown in areas such as image processing problems.

Summary:

mmWave radars have been successfully used for the detection, classification, recognition, segmentation, and tracking of mobile targets in the context of human-vehicle detection. The main objectives of the application of these radars in this field thus far have been:

- 1) To reduce the occurrences of vehicle-human or vehicle-animal collisions,
- 2) To obtain pedestrian, vehicular and other road user's statistics at traffic intersections,
- 3) To control the movement of vehicles and pedestrians at traffic lights or intersections,
- 4) Sensor fusion with cameras,
- 5) Driver assist systems.

TABLE 7. Summation of the various application areas, and the corresponding data processing pipelines used.

Area	Refs	Model	Contribution
Human vehicle recognition	Keegan <i>et al.</i> [22]	DSP	Monitors humans and vehicles at traffic intersections.
	Peter <i>et al.</i> [60]	DSP	Multiple pedestrian detection and tracking using a group tracking algorithm.
	Ren Yuan <i>et al.</i> [84]	SVM	Uses SVMs to classify interference waveforms.
	Xiangyu <i>et al.</i> [85]	DSP	Demonstrates the radar’s detection and ranging/angular localization capabilities through experimentation. AWR1642 BOOST + DCA100EVM
	Zihao <i>et al.</i> [58]	SVM	Investigates the performances of SVM classifier with different kernels: linear kernel, Gaussian kernel and polynomial kernel for human-vehicle classification.
	Feng <i>et al.</i> [80]	GMM	Performs unsupervised point cloud segmentation for pedestrians and vehicles.
	Khushi <i>et al.</i> [86]	DSP	Mechanically rotates the radar to determine the angle-of-arrival (AoA) of road targets.
	Patel <i>et al.</i> [87]	CNN	Employs deep learning on range-azimuth maps of radar targets.
Human detection, tracking & activity recognition	Ren Yuan <i>et al.</i> [28]	DSP	Utilises a Kalman filter to develop a radar and camera sensor fusion system for tracking objects.
	Zhao <i>et al.</i> [88]	CNN+LSTM	Combines CNNs and LSTM networks to track and identify humans from point cloud clusters.
	Maturana <i>et al.</i> [70]	3D CNN	Applies a 3D CNN on 3D voxels (volumetric occupancy grid) for object detection and recognition.
	Gambi <i>et al.</i> [78]	-	Provides raw data of people walking.
	Bai <i>et al.</i> [7]	dual-channel CNN	Compares dual-channel and single-channel CNNs for human gait recognition using joint time-frequency features.
	Zhang <i>et al.</i> [8]	CNN	Employs deep learning to classify human motion behaviour using micro-Doppler images.
	Singh <i>et al.</i> [26]	SVM, MLP, CNN & LSTM	Investigates different machine learning and deep learning models for human activity detection using radar point clouds.
	Liu <i>et al.</i> [27]	multi-layer CNN	Extracts features of the point cloud and passes them to the separate CNNs.
	Jin <i>et al.</i> [29]	CNN	Extracts the Doppler pattern from the point cloud data and apply deep learning to classify multiple patients behavior.
	Arganaras <i>et al.</i> [30]	DSP	Considers a shoe-mounted wearable radar for human gait analysis.
	Wang <i>et al.</i> [89]	CNN	Employs a line-kernel CNN on the base-band data to extract features for fall detection.
	Yang <i>et al.</i> [90]	CNN	Uses deep learning to identify the gaits of multiple users on range-Doppler profiles.
	Jin <i>et al.</i> [59]	RNN	Extracts the centroid of the point cloud data of the human body and applies a hybrid variational RNN to detect falls.
	Allam <i>et al.</i> [62]		
Wang <i>et al.</i> [68]	CNN+RNN	Combines a CNN and RNN for human activity recognition.	
Yu <i>et al.</i> [91]	DVCNN	Proposes a dual-view CNN and enhanced voxelization, along with data augmentation for human activity recognition.	
Gesture recognition	Ahmed <i>et al.</i> [32]	-	Presents a review of hand gesture recognition using radars.
	Palipana <i>et al.</i> [81]	LSTM	Extracts Spatio-temporal features from point clouds for hand gesture recognition.
	Li <i>et al.</i> [64]	3DCNN + LSTM	Tracks the motion of a finger to control the movements of a computer mouse.
	Yu <i>et al.</i> [92]	CNN + LSTM	Uses CNN and LSTM to classify hand gestures using range-angle images
	Goswami <i>et al.</i> [93]	ANN	Implements machine learning classifiers on the C67x DSP chip for gesture recognition of multiple users.
	Salami <i>et al.</i> [65]	K-NN	Develops a temporal K-NN for gesture recognition using point clouds.

TABLE 8. Summation of the various application areas, and the corresponding data processing pipelines used. *cont.*

Area	Refs	Model	Contribution
Patient vital signs monitoring	Alizadeh <i>et al.</i> [4]	DSP	Uses advanced phase unwrapping to extract the vital signs (respiration and heart rates) of a patient.
	Fioranelli <i>et al.</i> [9]	–	Discusses the application of radar for healthcare.
	Kebe <i>et al.</i> [31]	–	Provides a review of human vital signs detection using radars.
	Wang <i>et al.</i> [94]	DSP	Extracts the phase information of the intermediate frequency (IF) signals to determine the respiratory and heartbeat rates of a patient.
	Lohman <i>et al.</i> [95]	DSP	Uses Doppler information to determine the vital signs of a person.
	Ahmad <i>et al.</i> [55]	DSP	Uses phase information to extract the breathing and heart-rate of a person.
Pose estimation	Sengupta <i>et al.</i> [10]	GRU	Converts the point cloud data to voxels for skeletal pose estimation.
	Lu <i>et al.</i> [23]	LSTM	Uses LSTM and sensor fusion for ego-motion estimation.
	Almalioglu <i>et al.</i> [96]	DSP	Presents an IMU motion estimation system to estimate the six-degrees-of-freedom ego-motion of a moving radar.
	Sengupta <i>et al.</i> [97]	CNN	Uses a forked CNN to detect skeletal joints using radar point clouds.
	<i>et al.</i> [66]	NLP	Uses simulated mmWave radar like point-clouds to estimate the skeletal key-points of a human target.

Most works in the literature apply these radars to problems with a large intraclass and low interclass variation such as pedestrians, cyclists, cars and buses. In the future, it will be interesting to use mmWave radars to study classes with low intraclass variation such as distinguishing sedans, hatchbacks, and pick-up trucks. This is a task that cameras can perform effortlessly. However, we believe that the mmWave radar point cloud model can achieve impressive intraclass variation if it includes attributes that characterize the frequency and phase dependence.

B. HUMAN ACTIVITY RECOGNITION

While there are a lot of interesting application areas which were opened as a result of these mmWave devices, the current trend in the literature revolves around extracting the scattering centers of a target and applying deep learning algorithms to classify the reflections. One of the pioneering works in this field is presented by Zhao et al in [88]. The authors develop a mmWave radar system to track and identify people walking in non-line-of-sight radar conditions. In their setup, they assemble static obstacles (foam, wood, plastic and aluminium) and place them 1cm from the radar. A user walks towards the radar and then away from it. The radar collects all the scattering points from the scene and a deep neural network with Keras and Tensorflow is applied to identify and track the scattering points from the person. They were able to achieve a classification accuracy of 95%.

Apart from just identifying human beings in an environment, the authors in [8] used the same mmWave radar system to classify the behavior of people in real time. Their work focuses on generating standard micro-Doppler signatures from the range-Doppler sequence bins. Consequently, the generated micro-Doppler signatures were passed on to a convolution neural network to classify various activities such

as walking, swinging hands, and falling. With their method, they were able to get good prediction accuracies of over 94% for all classes. This study is particularly significant because it makes real-time human behavior classification using radars a reality.

Following satisfactory results in this area, some authors have further driven the application of mmWave radars into the domain of human gait and gesture recognition. This seemed particularly challenging in the past due to limitations in the radar system designs. Recent studies in the literature have focused on the use of the target's point cloud as input to the classifiers, which is a different line of action from the traditional micro-Doppler heat maps. For example, Singh et al. developed a system to recognize the activity of a human, based on the received point clouds [26]. Their RadHar framework is based on the IWR1443BOOST radar, and it focuses on recognizing activities such as boxing, jumping jacks, jumping, squats, and walking [26]. What sets them apart from other human gait recognition works is that they generate a voxelized data representation from the point clouds. The voxels are $(60 \times 10 \times 32 \times 32)$ in dimension and are passed on to the deep learning classifiers. Several classifiers such as SVM, MLP, Bi-Directional LSTM, and Time-distributed CNN are implemented using Sklearn and Keras. The performance comparison results reveal that the time-distributed CNN provides superior performance with accuracies of over 90% when combined with a bidirectional LSTM classifier.

Bai et al. compare two types of deep learning classifiers to evaluate their recognition accuracy for human gait analysis (normal walking, high leg walking, bending walking, normal jumping and one leg jumping) [7]. In their work, they propose a dual-channel deep CNN (DC-DCNN) and compare it with the most common single-channel deep CNN (SC-DCNN). According to the authors, they develop a symmetric network

which is composed of dual channels, and the input to such a network are 128×128 micro-Doppler images of human radar returns. Unlike the single-channel DCNN, the dual channels of the DCNN are made up of four convolutional layers, three pooling layers and a fully connected layer. With this setup, they were able to achieve an average accuracy of 91%, compared to an average accuracy of 80% achieved by a single channel deep CNN.

Summary:

Human activity recognition is an important research area with potential application in several fields, including but not limited to behavior detection, gesture recognition, and gait analysis. Earlier approaches have focused on using cameras and other invasive sensors to infer fine-grained human activities. However, the introduction of mmWave radars has shown great potential in this field due to high-resolution sensing. In that regard, several authors have reported impressive results in their research. On the other hand, it has been noted that the experiments were conducted in controlled environments. The human subject/s performed the activities directly in front of the radar, ($\varphi = 0^0$). In a realistic set-up, subjects could be at different angles from the radar, ($\varphi \neq 0^0$). Therefore, it is necessary to examine the effect of target orientation on the performance of the activity recognition algorithms. This area has received considerable interest in computer vision research [99]. As a result, we suggest that using mmWave radars for human activity recognition, where the subjects are at different azimuth angles, could be work for further research.

C. VITAL SIGNS MONITORING

Another interesting application area is the use of mmWave radars to remotely monitor the vital signs of a patient in a room. Typical vitals that can be measured using radar are the heartbeat per minute and the breathing rate. Extracting these parameters requires a key signal processing block that unwraps the phase of the signal, as shown in Figure 17. In fact, vital signs extraction using mmWave radars revolves around phase unwrapping of the signal. Several authors use this method, with minor modifications to some of the signal processing blocks.

Although this task had been initially attempted by the works presented in [95], the authors managed to obtain heart rate correlations between the reference sensor and the radar estimates of about 88%. This is mainly attributed to the radar systems in use at that time. They implemented a digital signal processor for Doppler radar in LabVIEW, and demonstrated that it is possible to remotely monitor the vital signs of a patient. However, this result was improved by the authors in [4], who demonstrated external sensor and radar correlations for heart rate measurements of approximately 94%. Here, the authors use a Texas Instruments mmWave (76-81 GHz AWR1443) FMCW radar to collect the raw data. In their method, they extract the phase from a range bin and measure the phase differences to extract the heartbeat pattern and the breathing pattern. These displacements are very small,

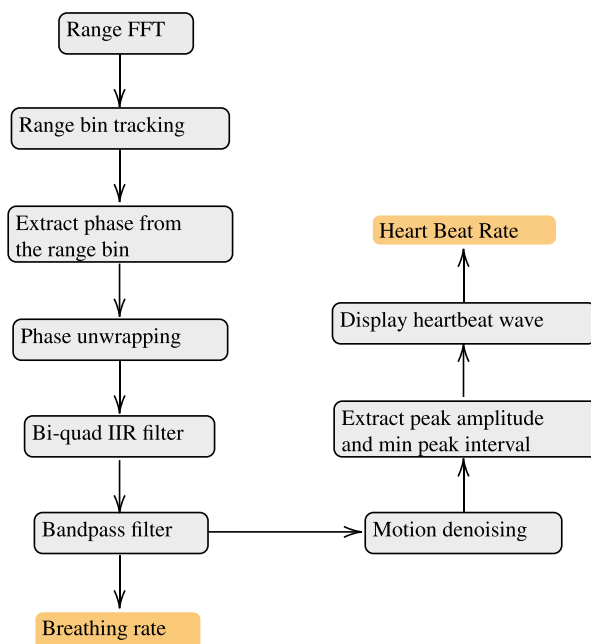


FIGURE 17. Vital signs detection workflow, adapted from [100].

in the order of 0.1–0.5mm and 1–12mm for heart rate and breathing rate respectively, but they manage to obtain heart rate correlations of 94%.

Similarly, the authors in [55] used the IWR1443 mmWave radar and the same processing chain to monitor the vital signs of a driver. However, the main difference between their work and the work presented in [4] is the set-up of the subject. Driver monitoring systems require the target to be seated on a chair, while patient monitoring systems require the target to be laying down. This presents challenges in the form of background clutter when a person is laying down [101]. This is because the clutter is more likely to be located in the same range bin as the signal to be measured, hence creating difficulties in measuring the change of phase of the required vital sign. However, this problem is much easier to handle when the clutter and the desired vital sign are located in different range bins. Reference [58] demonstrates the concept of range thresholding and clustering for noise suppression in mmWave radars. In their work, they use the scattering points from the AWR1642 mmWave radar to distinguish between vehicles and humans in a parking lot. To remove the reflection points from fixed clutter sources in their signal, the authors set a fixed range interval. Any points which are not in that interval are automatically removed. To overcome the problem of multipath, the authors implemented the DBSCAN [102] clustering algorithm, which is used to identify groups or clusters from the data. With this algorithm, high-density points are clustered together, while low-density points are treated as outliers, and are then eliminated from the data. As a result, this has motivated some authors to adopt range thresholding and clustering as a means of clutter and multipath suppression.

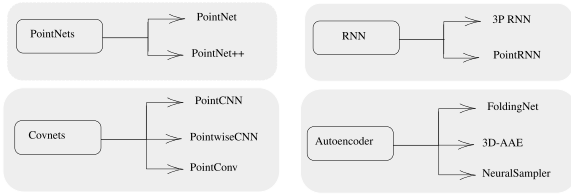


FIGURE 18. Raw point cloud processing algorithms. While this article focuses on techniques which have been applied to mmWave radar devices, it is important to understand the operation of other point cloud processing techniques since they have been around for a while. The progress made with deep learning algorithms on LIDAR point cloud data acts as a basis on which deep learning algorithms for mmWave radar point clouds are built, taking into consideration the main challenges of mmWave point clouds - sparsity and point instability.

Summary:

The objective of vital signs monitoring is to remotely acquire the subject’s pulse rate, and breathing rate using radars. Other vital signs monitoring tasks such as body temperature and blood pressure are, however, beyond the scope of the radar’s capabilities. In the literature, phase-unwrapping is an algorithm that has demonstrated positive results in the vital signs monitoring of a single subject. Algorithms that can determine the vital signs of multiple subjects in a scene are active research topics.

VI. DEEP LEARNING ON POINT CLOUD DATA

Mostly, deep learning approaches have been applied to several application areas such as computer vision [103], [104], speech recognition [105], and medical image analysis [106], among others. Until recently, a number of authors have pushed deep learning into the domain of LIDAR point cloud processing. This has resulted in the development of specialized deep learning pipelines that accept the raw point clouds, despite their unique characteristics. A summary of these algorithms is given in Figure 18. In what follows, we review the advances made in the field of deep learning on point cloud spectra. We particularly focus on those pipelines that have been adapted for mmWave radar point clouds, which we summarize in Tables 9 and 10. Two broad research questions will guide this section of the review: (i) Can we successfully apply ML/DL approaches directly on mmWave radar point clouds? (ii) What are the conditions that enable successful feature learning on mmWave point cloud data?

A. PointNet

PointNet, given by Charles et.al in 2016 is one of the pioneering deep learning architectures for processing raw point clouds [107]. Generally, this architecture was designed to overcome the challenges faced by conventional deep learning algorithms on point clouds. Therefore, it is suitable for tasks such as object classification, part segmentation, and semantic segmentation on an unstructured data set [107]. It achieved an overall classification accuracy of 89.2% on the ModelNet40 benchmark dataset compared to other volumetric deep learning architectures such as *VoxNet* and *3DShapeNets*,

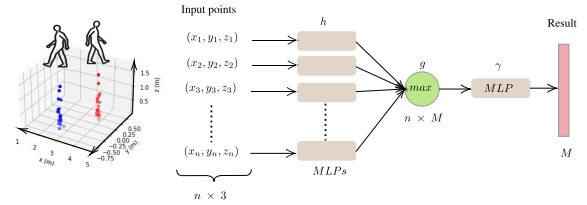


FIGURE 19. The PointNet architecture adapted from [107]. Key aspects of PointNet are that each 3D point, described by the (x,y,z) spatial coordinates, is treated separately by a shared MLP network, which is trained to extract different features of the input points. The architecture uses a single max-pooling strategy for down-sampling feature maps, thus controlling overfitting. Then the network selects points which carry more information based on learning optimization algorithms. The last layer is a fully connected layer which aggregates the data extracted from the previous layers to classify the point clouds into respective labels.

which achieved 85.9% and 84.7% respectively. As shown in Figure 19, a key aspect of PointNet and its derivatives is that they use a symmetric function to address the irregularity and invariance of the point clouds. Furthermore, the architecture uses a network of shared MLPs, a single max-pooling layer, and a fully connected layer. In particular, PointNet takes a flattened 1×3 vector and passes the input to a shared MLP with 5 hidden layers which are 64,64,64,128,1024 in size. 1024 features are extracted and the final MLP has 2 fully connected layers which are 256 in size.

Currently, a significant application of PointNet is in object classification, part segmentation and semantic segmentation on 3D point cloud data from depth sensors such as LIDAR [108]. Even though PointNet has proved successful in these areas, its application to solve radar problems is yet to be fully explored. Some attempts have been made to introduce this architecture to mmWave radar point clouds. Notably, Andreas et.al presented a 2D vehicle detection system in radar data with PointNets (based on an ARS 408-21 Premium radar sensor) [108] and Schumann et.al presented their work on semantic segmentation on radar point clouds [109]. However, there is no previous research in the literature on the use of PointNet-based deep learning frameworks specifically for mmWave radars with sparse point clouds.

Although PointNet achieves end-to-end learning on raw point clouds, a limiting issue of this algorithm and its derivatives is that the input point cloud is required to have the same size. This is a major challenge since the point cloud density from the radar is non-uniform on each frame. Discarding some of the data points or using spatial interpolation methods such as Inverse Distance Weighting (IDW) or Natural Neighbour (NN) may work for other point cloud tasks. In the case of mmWave radar applications such as human gait analysis, discarding points or interpolation may affect the point cloud distribution of the target class.

To address this issue, Yu et.al proposed an interesting approach for consuming the raw point clouds based on a CNN in 2020 [27]. The authors introduced their long-range gesture recognition system based on a TI mmWave radar. The algorithm was developed to recognize gestures such as

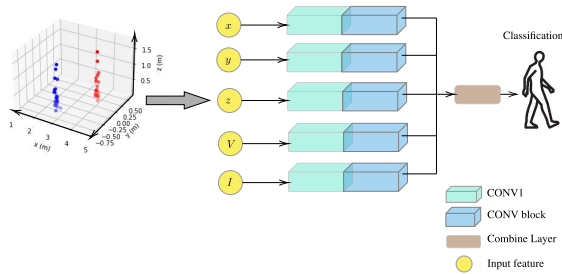


FIGURE 20. CNN architecture for mmWave point cloud data processing adapted from [27]. Five sets of point cloud features are passed on to a multi-branch convolutional network for learning the spatial and temporal dependencies.

knock, left swipe, right swipe, and rotate at a distance of about $2.4m$ from the radar (which is considered fairly long-range for such applications). The system incorporates three stages: signal transformation, information extraction, and lastly a Convolutional Neural Network. A notable feature of their work is the novel multi-branch CNN, which consists of two main blocks in sequence: CONV1 network and CONV block as shown in Figure 20. CONV1 network has a 7×7 convolutional layer with filter size 2×2 . The ReLu activation function is used after the layer. There is a 3×3 max-pool layer with a 2×2 filter at the end. The CONV block has a 3×3 convolutional layer which has a 2×2 filter. The feature values are aggregated to a combine layer which has a 3×3 convolutional layer, followed by a 1×1 filter. Finally, a fully connected layer with 65280 inputs is used to obtain the classification probabilities.

Their multi-branch CNN architecture has five layers, where each branch corresponds to a single feature of the 5D input point cloud: $x(m)$, $y(m)$, $z(m)$, Doppler (m/s), and intensity (dB) of each point. The network extracts the Spatio-temporal features of the point clouds which are then aggregated by the combine layer into a final classification result. Their architecture achieved a higher accuracy of 99% for the rotate gesture compared to other gestures such as knock 85.17%, left swipe 86.24%, and right swipe 82.17%. This is largely due to the benefits of the parallel convolution operation on each feature coordinate.

B. LONG SHORT-TERM MEMORY (LSTM)

Traditional COVNETs show their weaknesses when it comes to learning temporal dependencies in time-series data, such as features contained in successive range-Doppler images. Their architecture is not well suited for memorizing patterns in consecutive frames. On the other hand, LSTM networks, a class of Recurrent Neural Networks (RNNs), excel at learning the dependencies of data contained in image sequences. This is because the LSTM cells have loops which accept data from the preceding cell and the data of the current cell. In other words, they are capable of memorizing long sequences compared with classical feedforward neural nets, which are only capable of learning spatial dependencies. To make accurate predictions, the data sequences need to be learned in a specific order, *i.e.*, the order in which the data was received from the

target. As a result, LSTM nets need to know the position of each frame in the range-Doppler sequence such that the temporal correlations can be extracted for classification.

Similar to multilayer perceptron neural networks, LSTM networks also accept structured data. This means that the point cloud data must be transformed into images on a 2D grid to eliminate the problem of point cloud irregularity. This was demonstrated by Chris *et.al.*, who fused a TI AWR1843BOOST mmWave radar with an inertial measurement sensor for ego-motion estimation from consecutive radar point clouds [23]. The proposed system, *milliEgo*, consists of a two-layer LSTM network with 512 hidden units for temporal modelling. Spatial modelling is handled by three fully connected layers whose sizes are 128, 64 and 6 units respectively. The output of the last layer gives an estimate of the probability of the platform's 6 DoF pose. Although the authors report good accuracy values, they acknowledged that the point cloud sparsity has implications on computational efficiency, which in turn is vital for real-time applications.

Hybrid CNN and LSTM Networks:

Most mmWave radar classification problems have spatial features that are spread over time [5], [6], [7]. In that regard, spatial and temporal feature learning models are often integrated to provide superior classification performances compared to single deep learning architectures. A hybrid CNN and LSTM architecture is the most popular deep learning framework that has received considerable attention in the past. Results from previous research demonstrate satisfactory accuracy improvements. For instance, Akash *et. al* achieved an accuracy of 90% for a Time-distributed CNN + Bi-directional LSTM network, compared to 63% obtained for SVM classifier, 80% for a MLP, and 88% for a Bi-directional LSTM network [26].

Owing to these performance improvements, several authors have implemented this hybrid architecture in their frameworks: Peijun *et.al* tracking people with mmWave radar [88], Yasin *et.al* Ego-Motion Estimation [96], Almalioglu *et.al* Ego-Motion Estimation [110], Ziheng *et.al* Micro-gesture Cursor Input [64], Jih-Tsun *et.al* Hand Gesture Recognition [92], and Piyali *et.al* Real-Time Multi-Gesture Recognition [93].

In summary, a combination of feature extraction layers and sequence learning networks provides an architecture that is deep in both the spatial and temporal domains. They have great potential to address the classification requirements in classification problems that have subtle variations, such as human tracking, or identification and activity recognition using radar point clouds. However, results by Yan *et.al* indicate that LSTMs show their weaknesses when they are faced with very long sequences [111]. This is particularly challenging since higher frame rates result in high temporal resolution, which is necessary for extracting fine-grained temporal features. More recent research focuses on hybrid frameworks that comprise 1D CNNs and Temporal Convolution Networks (TCNs) to mitigate the problem of long sequences. They have been reported to outperform conventional

TABLE 9. Summation of TI's mmWave radar related projects. This list consists of authors who have used TI's mmWave radar products (AWR1642 BOOST, IWR1443BOOST and AWR1843BOOST) for different projects.

MAL	Ref	Theme	Architecture / layers	Number of classes	Feature (Dataset)	Test accuracy
	[114]	Adaptive Radar Density-Based Clustering and labeling	Clustering - Adaptive REDBSCAN	Kalman filtering		
	[58] & [115]	Human-vehicle classification	Kernel SVM: (Linear: Gaussian and Polynomial SVM)	(2) Pedestrians and vehicles at different speeds	$P_i = \{X_i, Y_i\}$	83% ped. & 87% for linear SVM, 80% pedestrian & 94% vehicle for Gaussian SVM and 83% pedestrian & 90% for Polynomial SVM.
SVM	[84]	Support Vector Machines for Classification of Automotive Radar Interference	SVM classification	Kalman filtering		
	[27]	Long-range gesture recognition	5 layer CNN network with 7×7 conv, 3×3 max pool layers with 2×2 strides	(4) Knock, left swipe, right swipe and hand rotate	$P_i = \{X_i, Y_i, Z_i, V_i, I_i\}$	$\approx 85\%$ for knock, left swipe and right swipe. 99% for hand rotate.
	[7]	Human gait recognition	Dual-channel DCNN: dual 4 conv layers, 3 pooling layers, and 1 fully connected layer	(4) Normal walking, one-legged walking and turn-back jumping	128×128 STFT images from the MOCAP dataset	96% for DC-CNN and 90% for SC-CNN with shorter sliding window.
	[8]	Real-time motion behaviour detection	2 CNN layers. Each layer with 3 conv2d, 3 Max Pooling, 4 Dropout, 4 leaky ReLU, 2 dense layers.	(4) Walking, swinging hands, standing/sitting and shifting	$P_i = \{X_i, V_i\}$ converted to microDoppler images	96% for walking, 99% for waving hands and 64% for sitting/standing.
CNN	[26]	Human activity recognition	SVM, MLP, LSTM, CNN	(5) Walking, jumping, jumping jacks, squats and boxing	$P_i = \{X_i, Y_i, Z_i, V_i, I_i\}$ converted to $10 \times 32 \times 32$ voxels	63% for SVM, 80% for MLP, 88% for Bi-directional LSTM and 90% for Time-distributed CNN + Bi-directional LSTM.
	[90]	Multi-user Identification Through Gait Analysis	CNN with 4 Conv layers, 4 max pool layers and a fully connected layer with 256 neurons	Identifying people	100×100 pixel rangeDoppler images	Correctly identifies 17 joints of a person and reconstructs a skeletal representation from the scattering points.
	[97]	Real-Time Human Skeletal Posture Estimation	Forked CNN; 3 layer CNN for each image channel and a 3 layer MLP	(4) Walking, Swinging left arm, Swinging right arm and Swinging both arms	The points $P_i = \{X_i, Y_i, Z_i, I_i\}$ are projected onto XY and XZ planes, then a 16×16 RGB image is constructed from the projections.	98.9% 3 layer CNN & 98.94% 4 layer CNN.
	[116]	Real-time multiple patients behaviour detection	Deep CNN: with 3 conv layers. Each conv layer with 3×3 kernel, leaky relu activation and 2×2 max pool layer	(6) Other behavior, walking, falling, swing hand for help, seizure and restless movement	$P_i = \{V_i\}$ converted to Doppler time images	
	[89]	A Millimetre-Wave Radar-Based Fall Detection Method	line kernel convolutional neural network (LKCNN)	Kalman filtering		

TABLE 10. Summation of TI's mmWave radar-related projects. This list consists of authors who have used TI's mmWave radar products (AWR1642 BOOST, IWR1443BOOST and AWR1843BOOST) for different projects.

MAL	Ref	Theme	Architecture /layers	Number of classes	Feature (Dataset)	Test accuracy
	[26]	Human activity recognition	SVM, MLP, LSTM, CNN	(5) Walking, jumping, jumping jacks, squats and boxing	$P_i = \{X_i, Y_i, Z_i, V_i, I_i\}$ converted to $10 \times 32 \times 32$ voxels	63% for SVM, 80% for MLP, 88% for Bi-directional LSTM and 90% for Time-distributed CNN + Bi-directional LSTM.
	[96]	Ego-Motion Estimation	RNN	Identifying people	100×100 pixel rangeDoppler images	Accurate trajectory error - $0.4m/s$ or $0.4rad/s$
LSTM	[111]	Ego-Motion Estimation	Bi-directional LSTM	(3) Pedestrian, car and clutter	$P_i = \{X_i, Y_i, Z_i, V_i, I_i\}$	
	[64]	Micro-gesture Cursor Input	3D Convolutional LSTM	Thumb-tip tracking	$P_i = \{X_i, Y_i, Z_i\}$	Mean Squared Error 16.4px or 1.35mm
	[59]	Fall Detection using 4D mmWave Radar	One hidden LSTN layer	Fall detection	$P_i = \{X_i, Y_i, Z_i, V_i, I_i\}$ → body centroid	Fall detection accuracy of 98%
	[23]	Egomotion Estimation	2 layer LSTM		$P_i = \{X_i, Y_i, Z_i\}$	1.3% 3D error drift
CNN + LSTM	[88]	Tracking people with mmWave radar	CNN+Bi-directional LSTM. CNN has 2 conv layers with 2 max pooling layers. LSTM with 128 hidden layers.	(12) Identification & tracking of 12 people	$P_i = \{X_i, Y_i, Z_i\}$	89% for 12 people
	[92]	Hand Gesture Recognition using Range-Angle Image	CNN + LSTM	11 gestures	Range-Angle images	
	[93]	Real-Time Multi-Gesture Recognition	ANN	6 hand gestures	$P_i = \{X_i, Y_i, Z_i\}$	8% DSP cycles
	[85]	Experiments with mmWave Automotive Radar Test-bed	Decision tree	(4) Walking, Swinging left arm, Swinging right arm and Swinging both arms	$P_i = \{X_i, Y_i, Z_i\}$	Decision Tree: 86.24% precision & 57.53% recall, CDMC: 92.03% precision & 96.82% recall.
	[80]	Point cloud Segmentation in Multimodal Traffic Monitoring	Multivariate Gaussian mixture model	(3) Pedestrian, car and clutter	$P_i = \{X_i, Y_i, Z_i, V_i, I_i\}$	89% for clutter, 61% car & 93% pedestrian
	[94]	Remote Monitoring of Human Vital Signs	Signal processing	(2) Breathing rate and heart rate	Phase extraction	93% classification accuracy of respiratory rate and heartbeat rate.
	[4]	Remote Monitoring of Human Vital Signs	Signal processing	Breathing rate and heart rate	Advanced phase unwrapping	94% and 80% sensor correlation for breathing rate and heart rate.
DSP	[22]	Traffic and intersection monitoring	Signal processing		Range-Doppler detection	-
	[28]	Radar camera fusion for detection and tracking	Signal Processing		$P_i = \{X_i, Y_i, Z_i\}$	0.29m range accuracy and 0.013rad in angular accuracy.
	[55]	Vital Signs Monitoring	Signal processing	(6) heartbeat and breathing rate	$P_i = \{V_i\}$ converted to Doppler time images	

sequence classification models such as RNNs when it comes to accuracy, memory, and computing resources [111], [112]. However, their attributes are yet to be tested on mmWave radar sequence applications.

In general, the scattering points from background clutter and multi-path have always been a challenge to deal with in mmWave radars [101]. Techniques such as range thresholding and clustering, which have been widely adopted by some authors for mitigating background clutter, are not always reliable due to the random nature of multipath. Unfortunately, this problem has rarely been addressed in the past. Only a few authors in the reviewed literature have explored new techniques for mitigating background clutter and multipath in single-chip mmWave radars.

VII. CONCLUSION

The main goal of the present study was to review the information available on the generation, collection, processing, and potential application areas of radar point clouds. Automotive mmWave radar devices were the specific focus. This paper has shown the state-of-the-art applications and processing frameworks applied on mmWave radar products, specifically in regard to radar point clouds as a data structure for representing target signatures. The review informs readers which radar point cloud processing pipelines were successful in solving the various detection, classification, recognition, and tracking problems. It also highlights the challenges and potential future directions of research. Additionally, this review serves as a reference guide for readers or innovators who want to build cutting-edge radar applications but do not have access to data collection equipment. By this, the following research questions have been addressed in this review:

- 1) What are radar point clouds and how are they generated?
- 2) What information do they carry and is it sufficient for feature learning?
- 3) What are the most efficient data processing frameworks for mmWave radar point clouds?
- 4) What are the application areas and why are they suitable for that task?
- 5) How can we simulate the point cloud data for various targets in the context of radar target detection, recognition, classification, and tracking?
- 6) What are the potential future directions of research?

REFERENCES

- [1] P. Yadav and S. Mutreja. (2022). *Automotive Radar Market Size, Share, Trends, Report by 2028*. [Online]. Available: <https://www.alliedmarketresearch.com/automotive-RADAR-market>
- [2] M. Series. (2015). *Systems Characteristics of Automotive Radars Operating in the Frequency Band 76–81 GHz for Intelligent Transport Systems Applications*. [Online]. Available: https://www.itu.int/dms_pubrec/itu-r/rec/m/R-REC-M.2057-1-201801-1!!PDF-E.pdf
- [3] F. Schwartau, S. Preussler, M. Krueckemeier, F. Pfeiffer, H. Stuelzebach, T. Schneider, and J. Schoebel, "Modular wideband high angular resolution 79 GHz radar system," in *Proc. 12th German Microw. Conf. (GeMiC)*, Mar. 2019, pp. 194–197.
- [4] M. Alizadeh, G. Shaker, J. C. M. De Almeida, P. P. Morita, and S. Safavi-Naeini, "Remote monitoring of human vital signs using mm-Wave FMCW radar," *IEEE Access*, vol. 7, pp. 54958–54968, 2019.
- [5] A. Ninos, J. Hasch, M. Heizmann, and T. Zwick, "Radar-based robust people tracking and consumer applications," *IEEE Sensors J.*, vol. 22, no. 4, pp. 3726–3735, Feb. 2022.
- [6] Y. Dong and Y.-D. Yao, "Secure mmWave-radar-based speaker verification for IoT smart home," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3500–3511, Mar. 2021.
- [7] X. Bai, Y. Hui, L. Wang, and F. Zhou, "Radar-based human gait recognition using dual-channel deep convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 12, pp. 9767–9778, Dec. 2019.
- [8] R. Zhang and S. Cao, "Real-time human motion behavior detection via CNN using mmWave radar," *IEEE Sensors Lett.*, vol. 3, no. 2, pp. 1–4, Feb. 2019.
- [9] F. Fioranelli, J. Le Kernec, and S. A. Shah, "Radar for health care: Recognizing human activities and monitoring vital signs," *IEEE Potentials*, vol. 38, no. 4, pp. 16–23, Jul. 2019.
- [10] A. Sengupta and S. Cao, "mmPose-NLP: A natural language processing approach to precise skeletal pose estimation using mmWave radars," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Mar. 1, 2022, doi: [10.1109/TNNLS.2022.3151101](https://doi.org/10.1109/TNNLS.2022.3151101).
- [11] M. I. Skolnik, *Radar Handbook*. New York, NY, USA: McGraw-Hill, 2008.
- [12] M. Arsalan, A. Santra, and C. Will, "Improved contactless heartbeat estimation in FMCW radar via Kalman filter tracking," *IEEE Sensors Lett.*, vol. 4, no. 5, pp. 1–4, May 2020.
- [13] C. Will, P. Vaishnav, A. Chakraborty, and A. Santra, "Human target detection, tracking, and classification using 24-GHz FMCW radar," *IEEE Sensors J.*, vol. 19, no. 17, pp. 7283–7299, Sep. 2019.
- [14] M. Jian, Z. Lu, and V. C. Chen, "Experimental study on radar micro-Doppler signatures of unmanned aerial vehicles," in *Proc. IEEE Radar Conf. (RadarConf)*, May 2017, pp. 0854–0857.
- [15] A. Santra, R. V. Ulaganathan, T. Finke, A. Baheti, D. Noppeney, J. R. Wolfgang, and S. Trotta, "Short-range multi-mode continuous-wave radar for vital sign measurement and imaging," in *Proc. IEEE Radar Conf. (RadarConf)*, Apr. 2018, pp. 946–950.
- [16] M. Arsalan and A. Santra, "Character recognition in air-writing based on network of radars for human-machine interface," *IEEE Sensors J.*, vol. 19, no. 19, pp. 8855–8864, Oct. 2019.
- [17] M. Stephan and A. Santra, "Radar-based human target detection using deep residual U-Net for smart home applications," in *Proc. 18th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2019, pp. 175–182.
- [18] Z. Zhang, Z. Tian, and M. Zhou, "Latern: Dynamic continuous hand gesture recognition using FMCW radar sensor," *IEEE Sensors J.*, vol. 18, no. 8, pp. 3278–3289, Feb. 2018.
- [19] J. Zhu, H. Chen, and W. Ye, "A hybrid CNN-LSTM network for the classification of human activities based on micro-Doppler radar," *IEEE Access*, vol. 8, pp. 24713–24720, 2020.
- [20] W. Ye, H. Chen, and B. Li, "Using an end-to-end convolutional network on radar signal for human activity classification," *IEEE Sensors J.*, vol. 19, no. 24, pp. 12244–12252, Dec. 2019.
- [21] W. Ye and H. Chen, "Human activity classification based on micro-Doppler signatures by multiscale and multitask Fourier convolutional neural network," *IEEE Sensors J.*, vol. 20, no. 10, pp. 5473–5479, May 2020.
- [22] K. Garcia, M. Yan, and A. Purkovic. (2015). *Robust Traffic and Intersection Monitoring Using Millimeter Wave Sensors*. [Online]. Available: <https://www.ti.com/lit/pdf/spyy002>
- [23] C. X. Lu, M. R. U. Saputra, P. Zhao, Y. Almaliglu, P. P. B. de Gusmao, C. Chen, K. Sun, N. Trigoni, and A. Markham, "MilliEgo: Single-chip mmWave radar aided egomotion estimation via deep sensor fusion," in *Proc. 18th Conf. Embedded Networked Sensor Syst.*, Nov. 2020, pp. 109–122, doi: [10.1145/3384419.3430776](https://doi.org/10.1145/3384419.3430776).
- [24] K. Ramasubramanian. (Dec. 2017). *Mmwave Radar for Automotive and Industrial Applications*. [Online]. Available: <https://www.readkong.com/page/mmwave-radar-for-automotive-and-industrial-applications-1240190>
- [25] C. Lovescu and S. Rao. (2015). *The Fundamentals of Millimeter Wave Radar Sensors*. [online]. Available: <https://www.ti.com/lit/pdf/spyy005>

- [26] A. D. Singh, S. S. Sandha, L. Garcia, and M. Srivastava, "RadHAR: Human activity recognition from point clouds generated through a millimeter-wave radar," in *Proc. 3rd ACM Workshop Millimeter-Wave Netw. Sens. Syst. (mmNets)*. New York, NY, USA: Association for Computing Machinery, 2019, pp. 51–56, doi: [10.1145/3349624.3356768](https://doi.org/10.1145/3349624.3356768).
- [27] Y. Liu, Y. Wang, H. Liu, A. Zhou, J. Liu, and N. Yang, "Long-range gesture recognition using millimeter wave radar," in *Green, Pervasive, and Cloud Computing*, Z. Yu, C. Becker, and G. Xing, Eds. Cham, Switzerland: Springer, 2020, pp. 30–44.
- [28] R. Zhang and S. Cao, "Extending reliability of mmWave radar tracking and detection via fusion with camera," *IEEE Access*, vol. 7, pp. 137065–137079, 2019.
- [29] F. Jin, R. Zhang, A. Sengupta, S. Cao, S. Hariri, N. K. Agarwal, and S. K. Agarwal, "Multiple patients behavior detection in real-time using mmWave radar and deep CNNs," in *Proc. IEEE Radar Conf. (RadarConf)*, Apr. 2019, pp. 1–6.
- [30] J. G. Argañarás, Y. T. Wong, R. Begg, and N. C. Karmakar, "State-of-the-art wearable sensors and possibilities for radar in fall prevention," *Sensors*, vol. 21, no. 20, p. 6836, Oct. 2021.
- [31] M. Kebe, R. Gadhafi, B. Mohammad, M. Sanduleanu, H. Saleh, and M. Al-Qutayri, "Human vital signs detection methods and potential using radars: A review," *Sensors*, vol. 20, no. 5, p. 1454, Mar. 2020.
- [32] S. Ahmed, K. D. Kallu, S. Ahmed, and S. H. Cho, "Hand gestures recognition using radar sensors for human-computer-interaction: A review," *Remote Sens.*, vol. 13, no. 3, p. 527, Feb. 2021.
- [33] S. A. Bello, S. Yu, C. Wang, J. M. Adam, and J. Li, "Review: Deep learning on 3D point clouds," *Remote Sens.*, vol. 12, no. 11, p. 1729, May 2020. [Online]. Available: <https://www.mdpi.com/2072-4292/12/11/1729>
- [34] R. Bhalla, J. Moore, and H. Ling, "A global scattering center representation of complex targets using the shooting and bouncing ray technique," *IEEE Trans. Antennas Propag.*, vol. 45, no. 12, pp. 1850–1856, Dec. 1997.
- [35] R. L. Moses, L. C. Potter, I. J. Gupta, and O. S. U. Columbus, "Feature extraction using attributed scattering center models for model-based automatic target recognition (ATR)," Ohio State Univ. Columbus, Columbus, OH, USA, Tech. Rep. ADA444563, Oct. 2005. [Online]. Available: <https://apps.dtic.mil/sti/citations/ADA444563>
- [36] C. Ozdemir, "Synthetic aperture radar," in *Encyclopedia of RF and Microwave Engineering*. Hoboken, NJ, USA: Wiley, 2005.
- [37] M. Wilmski, C. Kreucher, and J. Lauer, "Modern approaches in deep learning for SAR ATR," *Proc. SPIE*, vol. 9843, May 2016, Art. no. 98430N.
- [38] K. El-Darymli, E. W. Gill, P. Mcguire, D. Power, and C. Moloney, "Automatic target recognition in synthetic aperture radar imagery: A state-of-the-art review," *IEEE Access*, vol. 4, pp. 6014–6058, 2016.
- [39] T. D. Ross, J. J. Bradley, L. J. Hudson, and M. P. O'Connor, "SAR ATR: So what's the problem? An MSTAR perspective," *Proc. SPIE*, vol. 3721, pp. 662–672, Aug. 1999.
- [40] R. Bamler and P. Hartl, "Synthetic aperture radar interferometry," *Inverse Problems*, vol. 14, no. 4, pp. R1–R54, Aug. 1998, doi: [10.1088/0266-5611/14/4/001](https://doi.org/10.1088/0266-5611/14/4/001).
- [41] L. C. Potter and R. L. Moses, "Attributed scattering centers for SAR ATR," *IEEE Trans. Image Process.*, vol. 6, no. 1, pp. 79–91, Jan. 1997, doi: [10.1109/83.552098](https://doi.org/10.1109/83.552098).
- [42] E. F. Knott, J. F. Schaeffer, and M. T. Tuley, *Radar Cross Section*. Rijeka, Croatia: SciTech, 2004.
- [43] J. B. Keller, "Geometrical theory of diffraction," *J. Opt. Soc. Amer.*, vol. 52, no. 2, pp. 116–130, 1962.
- [44] Y. Akyildiz and R. Moses, "Scattering center model for SAR imagery," *Proc. SPIE*, vol. 3869, pp. 76–85, Dec. 1999, doi: [10.1117/12.373151](https://doi.org/10.1117/12.373151).
- [45] T. Long, L. Zhang, Y. Li, and Y. Wang, "Geometrical structure classification of target hrpp scattering centers based on dual polarimetric features," *IEEE Access*, vol. 7, pp. 141679–141688, 2019.
- [46] M. J. Gerry, *Two-Dimensional Inverse Scattering Based on the GTD Model*. Columbus, OH, USA: The Ohio State Univ., 1997.
- [47] D. R. Wehner, *High Resolution Radar*. Norwood, MA, USA: Artech House, 1987.
- [48] P. Monk, *Finite Element Methods for Maxwell's Equations*. New York, NY, USA: Oxford Univ. Press, 2003.
- [49] R. Bhalla and H. Ling, "Three-dimensional scattering center extraction using the shooting and bouncing ray technique," *IEEE Trans. Antennas Propag.*, vol. 44, no. 11, pp. 1445–1453, Nov. 1996.
- [50] K.-Y. Guo, X.-Z. Han, and X.-Q. Sheng, "Scattering center models of backscattering waves by dielectric spheroid objects," *Opt. Exp.*, vol. 26, no. 4, pp. 5060–5074, 2018.
- [51] C. Bourslier, N. Pinel, and G. Kubické, *Method of Moments for 2D Scattering Problems: Basic Concepts and Applications*. Hoboken, NJ, USA: Wiley, 2013.
- [52] C. Yin, B. Zhang, Y. Liu, and J. Cai, "A goal-oriented adaptive finite-element method for 3D scattered airborne electromagnetic method modeling," *Geophysics*, vol. 81, no. 5, pp. E337–E346, Sep. 2016.
- [53] J. M. Song, C.-C. Lu, and W. C. Chew, "Multilevel fast multipole algorithm for electromagnetic scattering by large complex objects," *IEEE Trans. Antennas Propag.*, vol. 45, no. 10, pp. 1488–1493, Oct. 1997.
- [54] K. Schuler, D. Becker, and W. Wiesbeck, "Extraction of virtual scattering centers of vehicles by ray-tracing simulations," *IEEE Trans. Antennas Propag.*, vol. 56, no. 11, pp. 3543–3551, Nov. 2008.
- [55] A. Ahmad, J. C. Roh, D. Wang, and A. Dubey, "Vital signs monitoring of multiple people using a FMCW millimeter-wave sensor," in *Proc. IEEE Radar Conf. (RadarConf)*, Apr. 2018, pp. 1450–1455.
- [56] F. Jin, "Millimeter-wave radar point cloud classification and interference mitigation," Ph.D. dissertation, Dept. Comput. Sci., Univ. Arizona, Phoenix, AZ, USA, 2020.
- [57] G. Zhang, X. Geng, and Y.-J. Lin, "Comprehensive mPoint: A method for 3D point cloud generation of human bodies utilizing FMCW MIMO mm-Wave radar," *Sensors*, vol. 21, no. 19, p. 6455, Sep. 2021.
- [58] Z. Zhao, Y. Song, F. Cui, J. Zhu, C. Song, Z. Xu, and K. Ding, "Point cloud features-based kernel SVM for human-vehicle classification in millimeter wave radar," *IEEE Access*, vol. 8, pp. 26012–26021, 2020.
- [59] F. Jin, A. Sengupta, and S. Cao, "mmFall: Fall detection using 4-D mmWave radar and a hybrid variational RNN autoencoder," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 2, pp. 1245–1257, Apr. 2022.
- [60] P. Nimac, A. Krpič, B. Batagelj, and A. Gams, "Pedestrian traffic light control with crosswalk FMCW radar and group tracking algorithm," *Sensors*, vol. 22, no. 5, p. 1754, Feb. 2022.
- [61] A. Badal and A. Badano, "Reducing the memory requirements of high resolution voxel phantoms by means of a binary tree data structure," *IEEE Trans. Radiat. Plasma Med. Sci.*, vol. 3, no. 1, pp. 76–82, Jan. 2019.
- [62] M. A. Ul Alam, M. M. Rahman, and J. Q. Widberg, "PALMAR: Towards adaptive multi-inhabitant activity recognition in point-cloud technology," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, May 2021, pp. 1–10.
- [63] P. Gong, C. Wang, and L. Zhang, "MMPoint-GNN: Graph neural network with dynamic edges for human activity recognition through a millimeter-wave radar," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2021, pp. 1–7.
- [64] Z. Li, Z. Lei, A. Yan, E. Solovey, and K. Pahlavan, "ThuMouse: A micro-gesture cursor input through mmWave radar-based interaction," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Jan. 2020, pp. 1–9.
- [65] D. Salami, R. Hasibi, S. Palipana, P. Popovski, T. Michoel, and S. Sigg, "Tesla-Rapture: A lightweight gesture recognition system from mmWave radar sparse point clouds," *IEEE Trans. Mobile Comput.*, early access, Feb. 23, 2022, doi: [10.1109/TMC.2022.3153717](https://doi.org/10.1109/TMC.2022.3153717).
- [66] A. Sengupta, F. Jin, and S. Cao, "NLP based skeletal pose estimation using mmWave radar point-cloud: A simulation approach," in *Proc. IEEE Radar Conf. (RadarConf)*, Sep. 2020, pp. 1–6.
- [67] Y. Sun, Z. Huang, H. Zhang, Z. Cao, and D. Xu, "3DRIMR: 3D reconstruction and imaging via mmWave radar based on deep learning," in *Proc. IEEE Int. Perform., Comput., Commun. Conf. (IPCCC)*, Oct. 2021, pp. 1–8.
- [68] Y. Wang, H. Liu, K. Cui, A. Zhou, W. Li, and H. Ma, "M-Activity: Accurate and real-time human activity recognition via millimeter wave radar," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 8298–8302.
- [69] Y. Xu, X. Tong, and U. Stilla, "Voxel-based representation of 3D point clouds: Methods, applications, and its potential use in the construction industry," *Autom. Construct.*, vol. 126, Jun. 2021, Art. no. 103675.
- [70] D. Maturana and S. Scherer, "VoxNet: A 3D convolutional neural network for real-time object recognition," in *Proc. IEEE/RSSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 922–928.
- [71] Z. Liu, H. Tang, Y. Lin, and S. Han, "Point-voxel CNN for efficient 3D deep learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–10.

- [72] X. Jiang, Y. Zhang, Q. Yang, B. Deng, and H. Wang, "Millimeter-wave array radar-based human gait recognition using multi-channel three-dimensional convolutional neural network," *Sensors*, vol. 20, no. 19, p. 5466, Sep. 2020.
- [73] K. Liu, Y. Zhang, A. Tan, Z. Sun, C. Ding, J. Chen, B. Wang, and J. Liu, "Micro-Doppler feature and image based human activity classification with FMCW radar," in *Proc. IET Int. Radar Conf. (IET IRC)*, 2021, pp. 1689–1694.
- [74] H. Ling, R.-C. Chou, and S.-S. Lee, "Shooting and bouncing rays: Calculating the RCS of an arbitrarily shaped cavity," *IEEE Trans. Antennas Propag.*, vol. 37, no. 2, pp. 194–205, Feb. 1989.
- [75] Z. Jianxiong, Z. Hongzhong, S. Zhiguang, and F. Qiang, "Global scattering center model extraction of radar targets based on wideband measurements," *IEEE Trans. Antennas Propag.*, vol. 56, no. 7, pp. 2051–2060, Jul. 2008, doi: [10.1109/TAP.2008.924698](https://doi.org/10.1109/TAP.2008.924698).
- [76] R.-C. Chou and S.-W. Lee, "Radar cross-section reduction studies of partially open cavity structures," *Int. J. Numer. Model., Electron. Netw., Devices Fields*, vol. 1, no. 4, pp. 207–220, Dec. 1988.
- [77] R. Bhalla, H. Ling, J. Moore, D. Andersh, S. Lee, and J. Hughes, "3D scattering center representation of complex targets using the shooting and bouncing ray technique: A review," *IEEE Antennas Propag. Mag.*, vol. 40, no. 5, pp. 30–39, Apr. 1998.
- [78] E. Gambi, G. Ciattaglia, A. De Santis, and L. Senigaglia, "Millimeter wave radar data of people walking," *Data Brief*, vol. 31, Dec. 2020, Art. no. 105996. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2352340920308908>
- [79] S. Z. Gurbuz, M. M. Rahman, E. Kurtoglu, T. Macks, and F. Fioranelli, "Cross-frequency training with adversarial learning for radar micro-Doppler signature classification (rising researcher)," *Proc. SPIE*, vol. 11408, pp. 58–68, May 2020.
- [80] F. Jin, A. Sengupta, S. Cao, and Y.-J. Wu, "MmWave radar point cloud segmentation using GMM in multimodal traffic monitoring," in *Proc. IEEE Int. Radar Conf. (RADAR)*, Apr. 2020, pp. 732–737.
- [81] S. Palipana, D. Salami, L. A. Leiva, and S. Sigg, "Pantomime: Mid-air gesture recognition with sparse millimeter-wave radar point clouds," *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 5, no. 1, pp. 1–27, Mar. 2021.
- [82] S. An and U. Y. Ogras, "Mars: MmWave-based assistive rehabilitation system for smart healthcare," *ACM Trans. Embedded Comput. Syst. (TECS)*, vol. 20, no. 5s, pp. 1–22, 2021.
- [83] D. Gusland, J. M. Christiansen, B. Torvik, F. Fioranelli, S. Z. Gurbuz, and M. Ritchie, "Open radar initiative: Large scale dataset for benchmarking of micro-Doppler recognition algorithms," in *Proc. IEEE Radar Conf. (RadarConf)*, May 2021, pp. 1–6.
- [84] R. Zhang and S. Cao, "Support vector machines for classification of automotive radar interference," in *Proc. IEEE Radar Conf. (RadarConf)*, Apr. 2018, pp. 0366–0371.
- [85] X. Gao, G. Xing, S. Roy, and H. Liu, "Experiments with mmWave automotive radar test-bed," in *Proc. 53rd Asilomar Conf. Signals, Syst., Comput.*, Nov. 2019, pp. 1–6.
- [86] K. Gupta, S. Joshi, M. B. Srinivas, S. Boppu, M. S. Manikandan, and L. R. Cenkeramaddi, "Localization of multi-class on-road and aerial targets using mmWave FMCW radar," *Electronics*, vol. 10, no. 23, p. 2905, Nov. 2021.
- [87] K. Patel, K. Rambach, T. Visentin, D. Rusev, M. Pfeiffer, and B. Yang, "Deep learning-based object classification on automotive radar spectra," in *Proc. IEEE Radar Conf. (RadarConf)*, Apr. 2019, pp. 1–6.
- [88] P. Zhao, C. X. Lu, J. Wang, C. Chen, W. Wang, N. Trigoni, and A. Markham, "MID: Tracking and identifying people with millimeter wave radar," in *Proc. 15th Int. Conf. Distrib. Comput. Sensor Syst. (DCOSS)*, May 2019, pp. 33–40.
- [89] B. Wang, L. Guo, H. Zhang, and Y.-X. Guo, "A millimeter-wave radar-based fall detection method using line kernel convolutional neural network," *IEEE Sensors J.*, vol. 20, no. 22, pp. 13364–13370, Nov. 2020.
- [90] X. Yang, J. Liu, Y. Chen, X. Guo, and Y. Xie, "MU-ID: Multi-user identification through gaits using millimeter wave radios," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Jul. 2020, pp. 2589–2598.
- [91] C. Yu, Z. Xu, K. Yan, Y.-R. Chien, S.-H. Fang, and H.-C. Wu, "Noninvasive human activity recognition using millimeter-wave radar," *IEEE Syst. J.*, vol. 16, no. 2, pp. 3036–3047, Jun. 2022.
- [92] J.-T. Yu, L. Yen, and P.-H. Tseng, "MmWave radar-based hand gesture recognition using range-angle image," in *Proc. IEEE 91st Veh. Technol. Conf. (VTC-Spring)*, May 2020, pp. 1–5.
- [93] P. Goswami, S. Rao, S. Bharadwaj, and A. Nguyen, "Real-time multi-gesture recognition using 77 GHz FMCW MIMO single chip radar," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Jan. 2019, pp. 1–4.
- [94] Y. Wang, W. Wang, M. Zhou, A. Ren, and Z. Tian, "Remote monitoring of human vital signs based on 77-GHz mm-wave FMCW radar," *Sensors*, vol. 20, no. 10, p. 2999, May 2020.
- [95] B. Lohman, O. Boric-Lubecke, V. M. Lubecke, P. W. Ong, and M. M. Sondhi, "A digital signal processor for Doppler radar sensing of vital signs," in *Proc. Conf. Proc. 23rd Annu. Int. Conf., IEEE Eng. Med. Biol. Soc.*, vol. 4, Jun. 2001, pp. 3359–3362.
- [96] A. Yasin and T. Mehmet. (Sep. 2019). *Milli-Rio: Ego-Motion Estimation With Millimetre-Wave Radar and Inertial Measurement Unit Sensor*. [Online]. Available: <https://paperswithcode.com/paper/milli-riego-motion-estimation-with>
- [97] A. Sengupta, F. Jin, R. Zhang, and S. Cao, "mm-Pose: Real-time human skeletal posture estimation using mmWave radars and CNNs," *IEEE Sensors J.*, vol. 20, no. 17, pp. 10032–10044, Sep. 2020.
- [98] S. M. Patole, M. Torlak, D. Wang, and M. Ali, "Automotive radars: A review of signal processing techniques," *IEEE Signal Process. Mag.*, vol. 34, no. 2, pp. 22–35, Mar. 2017.
- [99] M. Vrigkas, C. Nikou, and I. A. Kakadiaris, "A review of human activity recognition methods," *Frontiers Robot. AI*, vol. 2, p. 28, Nov. 2015.
- [100] S. Iyer, L. Zhao, M. P. Mohan, J. Jimeno, M. Y. Siyal, A. Alphones, and M. F. Karim, "mm-Wave radar-based vital signs monitoring and arrhythmia detection using machine learning," *Sensors*, vol. 22, no. 9, p. 3106, 2022.
- [101] J.-M. Muñoz-Ferreras, G. Wang, C. Li, and R. Gómez-García, "Mitigation of stationary clutter in vital-sign-monitoring linear-frequency-modulated continuous-wave radars," *IET Radar, Sonar, Navigat.*, vol. 9, no. 2, pp. 138–144, 2015.
- [102] D. Kellner, J. Klappstein, and K. Dietmayer, "Grid-based DBSCAN for clustering extended objects in radar data," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2012, pp. 365–370.
- [103] C. Ding and D. Tao, "Trunk-branch ensemble convolutional neural networks for video-based face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 1002–1014, Apr. 2018.
- [104] J.-H. Kim and C. S. Won, "Action recognition in videos using pre-trained 2D convolutional neural networks," *IEEE Access*, vol. 8, pp. 60179–60188, 2020.
- [105] P. Swietojanski, A. Ghoshal, and S. Renals, "Convolutional neural networks for distant speech recognition," *IEEE Signal Process. Lett.*, vol. 21, no. 9, pp. 1120–1124, Sep. 2014.
- [106] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway, and J. Liang, "Convolutional neural networks for medical image analysis: Full training or fine tuning?" *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1299–1312, Sep. 2016.
- [107] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Dec. 2017, pp. 652–660.
- [108] A. Danzer, T. Griebel, M. Bach, and K. Dietmayer, "2D car detection in radar data with PointNets," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 61–66.
- [109] O. Schumann, M. Hahn, J. Dickmann, and C. Wöhler, "Semantic segmentation on radar point clouds," in *Proc. 21st Int. Conf. Inf. Fusion (FUSION)*, Jul. 2018, pp. 2179–2186.
- [110] Y. Almalioğlu, M. Turan, C. X. Lu, N. Trigoni, and A. Markham, "Milli-RIO: Ego-motion estimation with low-cost millimetre-wave radar," *IEEE Sensors J.*, vol. 21, no. 3, pp. 3314–3323, Feb. 2021.
- [111] J. Yan, L. Mu, L. Wang, R. Ranjan, and A. Y. Zomaya, "Temporal convolutional networks for the advance prediction of ENSO," *Sci. Rep.*, vol. 10, no. 1, pp. 1–15, Dec. 2020.
- [112] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," 2016, *arXiv:1609.03499*.
- [113] R. Zhang and S. Cao, "Robust and adaptive radar elliptical density-based spatial clustering and labeling for mmWave radar point cloud data," in *Proc. 53rd Asilomar Conf. Signals, Syst., Comput.*, Nov. 2019, pp. 919–924.
- [114] Q. Wu, J. Chen, Y. Lu, and Y. Zhang, "A complete automatic target recognition system of low altitude, small RCS and slow speed (LSS) targets based on multi-dimensional feature fusion," *Sensors*, vol. 19, no. 22, p. 5048, Nov. 2019.
- [115] M. A. Koets and R. L. Moses, "Feature extraction using attributed scattering center models on SAR imagery," in *Algorithms for Synthetic Aperture Radar Imagery VI*, vol. 3721. SPIE, 1999, pp. 104–115.



HARRY D. MAFUKIDZE received the B.Sc. degree (Hons.) in physics from Midlands State University, Gweru, Zimbabwe, in 2009, and the M.Eng. degree in electronic engineering from the University of Stellenbosch, Stellenbosch, South Africa, in 2014. He is currently pursuing the Ph.D. degree in electrical engineering with the University of Cape Town, Cape Town, South Africa. He is also working with the Department of Applied Physics and Telecommunications, Midlands State

University. His research interests include radar signal processing, machine learning and deep learning, and developing mmWave radar point cloud processing frameworks.



JAN PIDANIC (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees from the University of Pardubice, in 2005 and 2012, respectively. His research interests include signal processing in passive radar systems, bistatic radars, clutter modeling, and optimization of signal processing algorithms with parallel processing techniques.



AMIT K. MISHRA (Senior Member, IEEE) received the Ph.D. degree from The University of Edinburgh. He is currently a Professor with the Department of Electrical Engineering, University of Cape Town (ranked at 157 by THE in 2021). Before Cape Town, he worked at Australia and India. He is also the Head at the Center for 5G in SDG, which works on projects related to the application of 5G in challenges related to sustainable development goals. He has successfully supervised nine Ph.D. students and holds five patents. His research interests

include radar system design and applied machine learning.



SCHONKEN W. P. FRANCOIS is currently working as a Senior Lecturer with the Radar Remote Sensing Group, University of Cape Town. His research interests include EM simulation and RF design.

...