POLITECNICO DI TORINO

Master's Degree in Mechatronic Engineering

Politecnico di Torino

Master's Degree Thesis

Review：mmWave Radar Point Cloud Processing Technology for Human Activity and Posture Recognition

Supervisors

Prof. Mihai T. LAZARESCU

Prof. Luciano LAVAGNO

Candidate

Chenglu Zhou

Academic year 2022-2023

Summary

In recent years, with the improvement of actual needs such as automatic driving, elderly health detection, and motion detection, the fields of human body detection, tracking, activity recognition, and gesture recognition have developed rapidly. In fact, light detection and ranging (LiDAR), cameras, and other radio technologies such as pulsed radio ultra-wideband (IR-UWB) and Wi-Fi have demonstrated their efficacy in these domains. Despite these promising advances and their potential for human tracking and activity recognition, a unified framework to assist these technologies is clearly lacking. The challenges posed by harsh signal propagation environments further widen this gap, complicating the tasks of person tracking and activity recognition [30], [31]. However, advances in mmWave radar and its integration with micro-Doppler signatures and point cloud data are reshaping the landscape, providing unprecedented capabilities for human-centric sensing.

This review thesis provides an in-depth study of state-of-the-art processing methods for point cloud data acquired by mmWave radar. We systematically study and elucidate the significant progress made in this field, focusing on fundamental keywords such as indoor localization, human tracking, human activity recognition (HAR), and human pose reconstruction.

This technical review will first provide an overview of how mmWave radar works and provide an in-depth look at the entire mmWave radar signal processing chain. Understand the principle and process of converting raw mmWave radar signals to the data form we need (such as point cloud, micro-Doppler signature).

Then we will start from the origin, development, sensors, data processing methods, and applications of point clouds, and introduce the development status of millimeter-wave radar point cloud signals, core aspects and challenges related to point cloud data. concern.

In this context, this thesis will deeply study the processing method of point cloud data collected by millimeter-wave radar, especially its practicability in the fields of human detection, tracking, human activity recognition (HAR) and human posture recognition. At this stage, this article will use the largest space to review a series of methods and algorithms for processing millimeter-wave radar point cloud data from the perspective of various input data.

In terms of method classification, we don't want to simply classify these methods with various topics, because after reviewing a lot of literature, we found that in many cases, such as human detection, tracking, human activity recognition (HAR) and human gesture recognition in most articles are located in the overlap. For example: when we realize human body posture recognition, we also realize human body activity recognition. We prefer to classify and compare

these methods from the representative characteristics of different methods designed by researchers. So, the classify and introduce each method will from the perspective of various data forms and get some interesting results from that as well.

The first category: Use the micro-Doppler feature as the input data.

Representative articles: "R. Zhang and S. Cao, "Real-Time Human Motion Behavior Detection via CNN Using mmWave Radar" The author proposes a method using micro-Doppler features converted from 3D point cloud information collected from radar systems integrated with CFAR algorithms, and This kind of data is used as a data set to train the convolutional neural network model, and finally realize the detection and classification of efficient human motion behavior.

The second category: Use voxelized point cloud data as input data.

First article been introduced is from Akash Deep Singh, Sandeep Singh Sandha, Luis Garcia, and Mani Srivastava. 2019. “RadHAR: Human Activity Recognition from Point Clouds Generated through a Millimeter-wave Radar.” This is an important work in the field of recognizing human actions or postures using millimeter-wave radar point cloud information. This study introduces RadHAR, the author defines this method as: "a framework designed for high-precision human activity recognition (HAR) using sparse and non-uniform point clouds." [3] The "sliding time window" and voxelization method is used by "RadHAR". The results achieved the best results at the time Among all the methods using millimeter-wave radar as a sensor, the best-performing deep learning classifier "Time-distributed CNN + Bi-directional LSTM" [3] designed by the author achieved an impressive A deep 90.47% accuracy. This demonstrates the efficacy of “RadHAR” in accurately identifying and classifying human activities.

The left work in this category: "P. Zhao et al., "mID: Tracking and Identifying People with Millimeter Wave Radar". In” mID” The authors introduce a system for human tracking and identification, termed “mID”, the advantage of “mID” lies in its ability to maintain high tracking accuracy without compromising visual confidentiality. Uniquely, as stated by the authors, this is the first instance of using point clouds generated by mmWave radar for tracking and identifying individuals as they walk. The performance of the system is remarkable, with a median position error of just 0.16 meters, and an identification accuracy of 89% for a sample of 12 people. These metrics underline the efficacy of the “mID” system in both tracking and identifying individuals based on their walking patterns.

Then C. Yu, Z. Xu, K. Yan, et al.: "Noninvasive Human Activity Recognition Using Millimeter-Wave Radar". In this work, they propose an accurate and efficient human activity recognition method based on augmented voxelization, data augmentation and dual-view machine learning. The results have demonstrated that the proposed system can achieve 97.61% and 98% accuracies during the

tests of fall detection and activity classification, respectively.

After compared the above-mentioned several voxelization methods and found that the combination of enhanced voxelization method and Data Augmentation used in the article "Noninvasive Human Activity Recognition Using Millimeter-Wave Radar" finally achieved the best results. The reason Because random voxelization has the advantages of improving data calculation efficiency, orderly data storage and down-sampling, it is beneficial to extract multi-scale and multi-level local feature information, and it is conducive to maintaining the spatial correlation of point clouds, but the voxelization process is inevitable. It is still a big challenge to cause information loss, but the author of this article successfully avoided this problem by using Data Augmentation. At the same time, we also found that using the difference model and the fusion model can significantly improve the speed of data processing and the accuracy of classification results.

The third category: Use multi-dimensional point cloud data as input data.

First Article: Arindam Sengupta, Feng Jin, et al.: “mm-Pose: Real-Time Human Skeletal Posture Estimation using mmWave Radars and CNNs”. This study introduces a novel real-time technique for estimating and tracking human skeletal structures, employing millimeter-wave (mmWave) radar and convolutional neural networks (CNNs). According to the authors, this approach is pioneering, being the first to successfully detect more than 15 distinct skeletal joints using mmWave radar reflection signals. In terms of performance, the method has shown promising results, yielding average localization errors of 3.2 cm in depth (X-axis), 2.7 cm in elevation (Z-axis), and 7.5 cm in azimuth (Y-axis). This innovative approach promises to advance the field of human skeleton tracking using non-invasive technologies.

Then is the article from Meng, Z., Fu, S. et al. “Gait Recognition for Co-Existing Multiple People Using Millimeter Wave Sensing.” The authors present an innovative deep-learning oriented approach for recognizing human gait patterns using millimeter-wave (mmWave) technology, termed mmGaitNet. This system falls under the umbrella of Multi-channel Attribute Deep Networks.Significantly, mmGaitNet showcases impressive accuracy rates in various scenarios. It reaches 90% accuracy when identifying a single person and maintains 88% accuracy even in complex scenarios with five co-existing individuals. In comparison, the existing methods have proven to be less effective, with accuracy levels not surpassing 66% in either of the aforementioned scenarios. This robust performance underlines mmGaitNet's potential as a leading solution in the field of mmWave gait recognition.

Next from Jiang, X.; Zhang, Y. et al. “Millimeter-Wave Array Radar-Based Human Gait Recognition Using Multi-Channel Three-Dimensional Convolutional Neural Network.” In this study, they explore two fundamental issues related to human gait detection using radar, namely, classification and recognition of human gait patterns. Then present a novel method based on millimeter-wave array radar,

and developed a multi-channel three-dimensional convolutional neural network (CNN), which is an enhancement of the existing residual network model. This model is specifically designed for classifying and recognizing human gait by employing hierarchical extraction and fusion of multi-dimensional features. The inputs for the network are the three-dimensional coordinates, speed of movement, and intensity of strong scatter points during target motion. This multi-channel convolution method effectively extracts motion features and facilitates the classification and recognition of typical daily actions such as walking and jogging. In terms of performance, their experimental results have been highly promising, achieving over 92.5% recognition accuracy for common gait categories like jogging and normal walking. This highlights the effectiveness of the proposed method in identifying and classifying different types of human gait.

The last article is from Sizhe An and Umit Y. Ogras. 2021. “MARS: mmWave-based Assistive Rehabilitation System for Smart Healthcare." The MARS system is capable of reconstructing 19 human joints and their corresponding bone structures from point cloud data generated by millimeter wave (mmWave) radar. In terms of result evaluation, the author uses the "mean absolute error" to evaluate the performance of the system, and the obtained results show that the "mean absolute error" is 5.87 cm, which is a gratifying result. This proves that the "MARS" system has a very powerful potential in effectively reconstructing human joints and bone structure, that is, a more accurate human posture, after accurately calculating the positions of key points of human bones based on the point cloud data generated by millimeter-wave radar. Even when dealing with complex and demanding rehabilitation sports.

Then I analyzed and compared "mm-Pose" and "MARS", "mmGaitNet" and "MC-3DCNN". The conclusion is that compared with "mm-Pose", using the same data set and using the same feature map, "MARS" achieves better than "mm-Pose" under the condition of reducing the complexity of the model. "Better performance. In terms of hardware settings, "mmPose" requires two radars, while "MARS" only uses one radar, making it more practical and easier to use. In addition, MARS can handle complex rehabilitation exercises because rehabilitation exercises require more precise results and faster response speeds, while mmPose can analyze joint movement conclusions during walking to help medical staff assess the patient's status.

For "mmGaitNet" and "MC-3DCNN". "mmGaitNet" requires placing two radars diagonally for maximum recognition accuracy. At the same time, if the number of people in an open space increase, the recognition accuracy will drop rapidly in the case of a per capita gathering. For "MC-3DCNN", although it has reached an average accuracy rate of 93%, the system itself is designed for single-person gait recognition and is not suitable for deployment in open spaces. Yes, the usage scenarios of the system are very limited. It is only suitable for use in spaces such as single wards. Therefore, in terms of gait recognition, research still has a lot of room for improvement.

At this point, we can make a brief summary of the subject of human activity recognition. In essence, human activity recognition based on 3D point clouds is a classification problem. At this stage, the accuracy of human activity recognition in the case of a single person has reached 97%. [15]. According to the different data input, we can divide these methods into the four categories mentioned above, but we look at the models they use again and we can find that, for 3D point cloud data and point cloud data conversion When voxelizing incoming voxel data, researchers often use combined models based on CNN and LSTM for design. At the same time, in order to cope with the large number of features carried by multi-dimensional point cloud data, decomposing these features in the form of difference is also a very effective way to improve the model processing ability. practice. For micro-Doppler data, articles [1] and [15] prove the outstanding ability of the CNN model in this regard.

The fourth category: use point cloud and range Doppler as the input of the fusion model

The article from Huang, Y.; Li, W. et al. “Activity Recognition Based on Millimeter-Wave Radar by Fusing Point Cloud and Range–Doppler Information.” The researchers implement a hybrid model that combines Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) to extract time-sequential features from point clouds. Additionally, a separate CNN model is used to capture features from range-Doppler data.The evaluation of this combined method, based on the dataset used, revealed that it achieves superior accuracy compared to approaches that utilize either type of information individually. The recognition accuracy of the combined method was an impressive 97.26%. This marks a roughly 1% improvement over networks that rely on only one type of data input, underscoring the effectiveness of combining different data types and models for more accurate results.

In the part of human pose recognition. Some representative methods, such as the most representative research direction, are to reshape human pose through skeletal joint localization. This includes the aforementioned "mm-Pose" and "MARS," which also use skeletal positioning to recognize human activity. A brief introduction to the method is also found in other works such as: H. Cui and N. Dahnoun, "Real-Time Short - Human Pose Estimation Using Millimeter-Wave Radar and Neural Networks", and Kong, Xiangyu Xu et al. M3Track: mmWave-based multi-user 3D pose tracking".

While exploring the field of human pose estimation, it becomes apparent that there is a plethora of models and methods available. However, one significant challenge is the limited availability of open-source research databases akin to those for lidar and cameras. Researchers often have to invest substantial effort in designing experimental scenarios and hardware equipment, which can be quite time-consuming.

In the second chapter, a novel solution to this challenge is presented:

"mmPose-NLP". This approach, based on natural language processing (NLP), is utilized for estimating skeletal poses using simulated millimeter-wave (mmWave) radar point cloud data. And then the author uses the simulated human skeleton key point millimeter-wave radar point cloud data to train the deep learing model and finally achieved a significant result.

Following this, the work of Sizhe An and Umit Y. Ogras is highlighted, who developed a quick and adaptable framework for human pose estimation. Their article, "Fast and scalable human pose estimation using mmWave point cloud," details a baseline model composed of a Convolutional Neural Network (CNN). After fine-tuning the FUSE model for just 5 epochs, it achieves a Mean Absolute Error (MAE) of 8.3 cm, which is 1.3 cm lower than the baseline. This demonstrates how efficient training strategies can enhance the performance of machine learning models in this field.

"MARS" as the best skeletal key point estimation method introduced before. I make a compare between "MARS" and "FUSE". FUSE achieves a lower mean absolute error (MAE) in joint coordinate estimates, showing a 34% improvement over "MARS". Additionally, "FUSE" is able to adapt to unseen scenarios within five epochs, which is four times faster than "MARS".

According to the result FUSE is a better model than MARS for mmWave point cloud-based human pose estimation due to its improved point cloud representation and the incorporation of meta-learning. These enhancements result in higher accuracy and faster adaptation to new data, making FUSE a more versatile and efficient model for human pose estimation tasks.

At this point, we can make a summary for human body pose recognition. At this stage, our most effective and accurate solution in human body pose estimation is to use computer-simulated radar point clouds to locate bone key points. However, such a method will face great challenges in the selection of noise and the stability of the model. However, different from human activity recognition, reshaping human posture through bone key point positioning is essentially a regression problem. The purpose of the experiment is to obtain more accurate human bone position information, so the methods that can be applied in this regard are also more diverse. It also has more potential for research. But what we can still see is that although all methods have designed different models, we can still see that many methods use CNN or LSTEM models as baseline models [11], [13] or use CNN's Some modules [12], from which we can also observe that the outstanding spatial and temporal interpretation capabilities of the CNN and LSTM models in human activity recognition still have great advantages in the application of human pose recognition. At the same time, we can also consider some future research possibilities. For example, there is a lack of open-source databases available. In addition, the author of the article [8] adopted the "seq2seq" natural language model, so now that various natural language models are blowing out, is there any other model that can get better results?

But at the same time, what we can think about is that the essence of human

activity is the constantly changing posture of the human body, so whether it is possible to use the method of skeletal key point positioning and reshaping to identify human activity.

The increasing importance of millimeter-wave (mmWave) radar has had a significant impact on various research fields, especially those related to human detection and tracking. However, the interpretation of point cloud data collected by mmWave radar, a key element in these processes, remains a challenging task. It can be noticed that there are already many researchers using millimeter-wave radar to design many methods with very high accuracy in the fields of human body tracking, human activity recognition and human gesture recognition. But it is worth noting that most of these methods have more restrictions, such as the performance requirements of the hardware and the settings of the radar. Moreover, the point cloud obtained based on the existing radar system is still very sparse, which has great limitations for future research.

Table of Contents

Chapter I

Introduction

Nowadays, our society is becoming more and more intelligent and digital, especially in the field of traditional monitoring, traditional camera monitoring has the best effect, but now we gradually realize the importance of privacy protection, but cameras are considered to be in the privacy protection, and low light conditions are very limited. Therefore, with the gradual maturity of lidar, WiFi, infrared camera, radar and other technologies, people began to try to develop applications in related fields of these devices.

Among these sensors, millimeter-wave radar is considered to be a device with great market application potential, because it is relatively less restricted by light, environment, and venue, and at the same time has very high privacy and confidentiality. Therefore, in the future, it will be widely used in medical, health, military, public places and automatic driving.

The goal of human location tracking and activity recognition technology is to create real-time personnel protection that adapts to any indoor environment or semi-open environment. Regardless of the composition of this open or semi-open space, this makes the technology adaptable to different application scenarios, such as personnel security in office environments or factory floors, or anti-theft security in home environments with limited space, or rehabilitation needs the health detection of patients, and the elderly can play a very critical role even in emergency rescue.

It is based on these facts that this work initiates this work, which aims to present an existing representative analysis of point cloud data collected using mmWave radar as a sensor in the fields of human localization, human activity recognition, and human pose recognition technology. In addition, specific analysis, interpretation, and comparison of the advantages and disadvantages of existing technical methods with similar results are carried out. It is hoped that future researchers will be provided with a simple and fast path to understand the current state of development in this field. However, it should be pointed out that: this work is not to screen the optimal solution at the emergence stage, but to sort out technical methods and provide innovative ideas for subsequent researchers.

Now I introduce the chapter structure of this article, as follows:

In the second chapter, this work will provide an overview of the radar signal processing chain, mainly introducing the development of radar, the basic working principle of FMCW radar system, and the processing process of radar to generate

other waveforms and point cloud signals. Raw radar signal.

Chapter three introduces point clouds. The article will start with "What is a point cloud?". Then the basic point cloud data processing flow is introduced.

This chapter will focus on more than ten representative techniques in this field, such as: methods based on traditional micro-Doppler signals [1], methods based on voxelized point clouds [3], [16], and multidimensional point cloud data [5], [7], [10], and the method of using micro-Doppler features and point cloud signals [15], and the method of simulating human point cloud signals by simulation [8].

In the article, I divided them into two categories based on the experimental goals, one is human activity recognition, and the other is human gesture recognition. Because human body activity recognition is a classification problem in terms of problem classification, human body point cloud information will be collected in the experiment, and then the recognition result of human body activity will be obtained through the analysis of point cloud information, while human body posture recognition is a regression problem, usually we Human body pose is reconstructed using key points of human skeleton.

At the end of the two categories of human activity recognition and human gesture recognition, the methods in the two branches are compared in the form of Table XI and Table XIII. The technical characteristics of each technical method are compared in this work. This article first introduces the experimental settings in the technical methods, and then gradually introduces the technical process, experiments and verification results. This article will make a comprehensive comparison of the technical methods introduced at the end of each type of technology. The following comparison will be carried out from several aspects.

i. Comparison of the experimental results of the methods.
ii. The stability of the comparison method results.
iii. Comparing the adaptability and universality of the methods in different environments.
iv. Complexity.
v. The hardware or experimental cost of the methods being compared.

Second, when comparing similar methods, this work will make basic distinctions based on the subject matter of the methods. Although some articles are classified into one category from certain angles, the results are not comparable afterwards. For the former example, the technique method "mm-Pose" [4] and the technique method "mmGaitNet" [5].

Since the results of various methods have been described and compared in detail in the text, therefore, in the two summary tables Table XI, the method of human activity recognition is analyzed from the sensors used, the data form, and the deep learning model type, etc. aspects were compared. Combining the result data we obtained from the references, we can get the classification results. For human activity recognition in the case of a single person, it is easier to obtain

better results by using multiple forms of input data and differential models, for the choice of baseline model from the results, CNN and LSTM are the more commonly used baseline models and the two models with the best results. The reason is that these two models have outstanding capabilities in spatial analysis and time analysis respectively, so in It has a very good effect on human activity recognition. But at the same time, we also see that although 97% recognition accuracy can be achieved in single-person scenarios [15], based on our knowledge, the number of models suitable for multi-person scenarios is still limited and the types of activities are very limited. It is limited to gait recognition [], and when the number of human targets increases, the recognition accuracy will drop greatly. Moreover, in the entire field of human activity recognition, the recognition accuracy for relatively low-movement sports is relatively low, and these are directions for further research in the future.

At the end of the human body pose recognition, Table XIII is also used to summarize the table, comparing the sensors used, the data format, the number of key points of the bones, and the accuracy. In this field, the bones we observed the method of using simulated data in the key point positioning method [8] has obtained the best results in the experiment. Compared with other methods that use the point cloud data collected in the experiment, this method is more expensive in terms of experimental cost and accuracy. The effect is very good, but the setting of the noise level is very difficult, which will cause great challenges to the results and stability of the model. At the same time, it is difficult to determine how adaptable the method is to different environments and goals. In addition, due to the emergence of a large number of NLP language models this year, many different NLP models can be applied to this method, so this direction is worthy of further digging.

In general, at this stage, we have made great progress in the field of human activity recognition and human pose recognition, but there is still a lot of room for improvement and many problems to be studied. At the same time, the technical development of radar itself restricts the development of this field. The typical problem is that the radar point cloud has high sparsity, which poses great challenges to subsequent experiments, especially when dealing with activities with indistinct motion characteristics.

Chapter II

Radar

2.1 Overview of the Radar Signal Processing Chain

This section provides a concise exploration of the procedures involved in radar signal detection, with a particular focus on range and velocity estimation in various mmWave radar systems. These systems include Frequency Modulated Continuous Wave radar, Frequency-Shift Keying (FSK), which are frequently used in automotive radars and indoor localization radars. A depiction of a typical radar system is presented in Figure.1 for further reference.

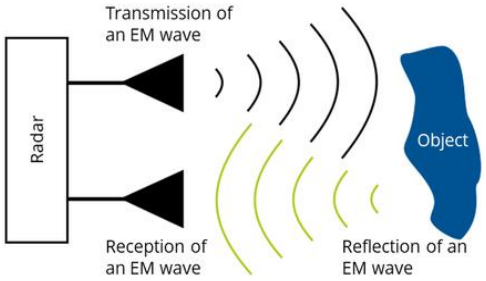

Figure (1): Radar basic working principle: The object is detected by comparing the transmitted (TX) and the received (RX) electromagnetic wave.[96]

2.2 Frequency Modulated Continuous Wave (FMCW) Radar

Radio detection and ranging (radar) technology was first developed in the 19th century. But it wasn't until 1940 that time-band millimeter-wave radar technology was first released. It was first applied to marine navigation, but its development was greatly restricted due to its low power and large transmission loss. In the mid-1970s, Germany's AEG-Telefunken and Bosch began to invest in research on the application of millimeter-wave radars in automobiles for collision avoidance. Due to the high cost, the development of millimeter-wave radars stagnated. Until the early 1980s, many famous universities, research institutes and enterprises all over the world joined the upsurge of researching millimeter-wave radar, which directly promoted the rapid development of millimeter-wave technology. In the late 1980s, the "European Efficient and Safe Traffic System Plan" launched the research program of vehicle-mounted millimeter-wave radar again, and millimeter-wave radar technology has entered a period of explosion since then. After entering the 1990s, millimeter-wave radar car anti-collision technology gradually matured, and millimeter-wave radar really

entered commercial use. The most widely used area is in the automotive field.

The main function of the millimeter-wave radar is to detect objects and calculate the parameters of the detected objects through the radar system, such as: speed, distance, and azimuth. The radar system's time delay and Doppler frequency estimates determine the accuracy of the radar data.

There are many types of radar, which can be classified according to different standards.

According to the wavelength, it can be divided into long-wave radar and short-wave radar. The wavelength of long-wave radar is meter or decimeter level, its resolution is low, but its penetration is strong, and it is generally used for broadcasting, military early warning and satellite communication. The wavelength of short-wave radar is centimeter or millimeter level, its resolution is high, but its penetration is poor, and it is generally used for surveying and mapping, short-range communication and vehicle applications.

According to the waveform, it can be divided into pulse radar and continuous wave radar. Pulse radar uses the time difference between pulse transmission and reception to determine the distance of the target and cannot measure the speed of the target. This principle is very similar to LiDAR. The transmitted signal of the continuous wave radar is continuous in time, and the frequency of the transmitted signal changes with time, so it also becomes a continuous frequency modulation wave. What this article is going to introduce is the millimeter-wave radar based on continuous frequency modulated wave (FMCW). Figure.2 provides a basic description of the FMCW system through a block diagram.

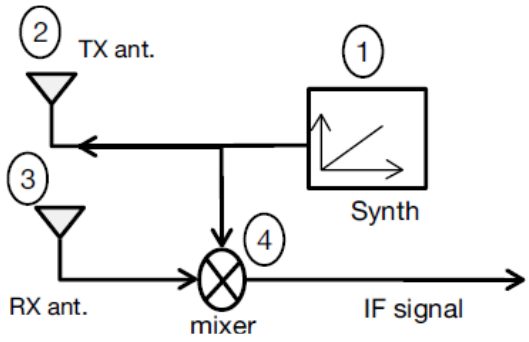

Figure (2): Block diagram of a FMCW radar. The synthesizer is responsible for the chirp generation, the mixer mixes received (RX) and transmitted (TX) obtaining in output the intermediate frequency (IF) signal, which is analyzed to find target(s).[51]

The operation of an FMCW system, which transmits sequences of a Linear Frequency Modulated (LFM) signal, also referred to as a chirp signal. This kind of signal increases linearly with time, over a bandwidth range that can reach up to 4 GHz and a carrier frequency range of 76–81 GHz [52]. The core principle of radar systems involves the transmission of an electromagnetic signal that is then reflected by objects in its trajectory. Specifically, in FMCW radars, a signal is utilized wherein the frequency escalates linearly with time - a signal type also known as a chirp show in Figure.3. From the plot the chirp is characterized by

"fc" which is start frequency, "B" which is bandwidth, "Tc" means duration, and the slop "s" control the rate of frequency of the chirp.

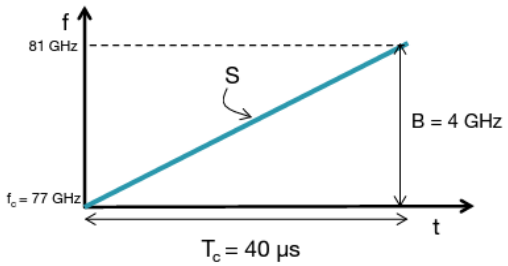

Figure (3): Chirp signal, with frequency as a function of time.[51]

The system then captures the reflected signals bouncing back from the targets. Once the moving object with respect to the radar been detected, the Doppler effect with take place show in Figure.4 at the receiving end, these received signals are mixed with the transmitted (chirp) signals in a mixer the Figure.2 shows with Block diagram. The Doppler effect then induces a frequency shift in the received signal, correlating to the radial velocity of the target. This will result in a heightened frequency if the target is advancing towards the radar, while a receding target will yield a lower frequency.

Figure (4): Radio wave signal which changes frequency due to Doppler effect caused by the movements of the detect moving objects.[97]

The operational mechanism of FMCW radars, illustrated in Figure.5, showcases the key parameters: "Δt", which represents the time difference between the transmitted and received signals; "Δf" the discrepancy in frequency; and "f_D", the frequency shift instigated by the Doppler effect.

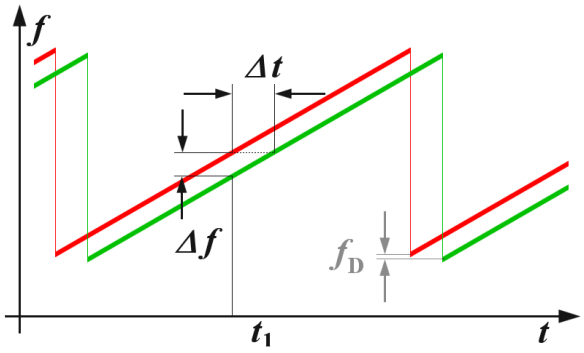

Figure (5): Operational mechanism of FMCW radars to Doppler effect.

Come back to the FMCW system. As the block diagram shows in Figure.2 received signals are mixed with the transmitted (chirp) signals in a mixer. The result of the mixer, an intermediate frequency (IF) signal is produced. Performing a Fourier transform on this signal yields a beat signal. The frequency of this signal remains consistent and is equal to S * τ, where τ denotes the time delay between the transmitted and received signals. This frequency, as depicted in Figure.6 is utilized to determine the range of the detected target. If multiple targets are detected, the IF signal will consist of several sinusoids, each having a unique frequency corresponding to a specific target [51].

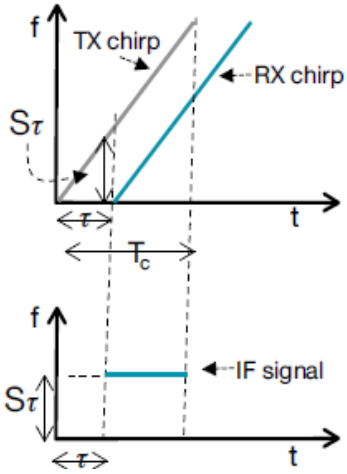

Figure (6): Intermediate frequency (IF) resulting from the mixer output, a signal with constant frequency.[51]

Also, this process results in a new frequency known as the beat frequency signal, calculated as follows:

$$f_b = 2ds/c \tag{1}$$

where "b" is the distance of the object from the radar, "s" is the slope of the chirp signal, and "c" is the speed of light.
And then we can calculate the range of the target detected by the radar with the function blow:

$$R = \frac{(C \cdot T_c \cdot f_b)}{(2 \cdot B)} \tag{2}$$

where "R" respect to the target distance, "c" has introduced before the speed of light, "fb" is beat frequency corresponding to the target, "B" means bandwidth of the signal.

We have measured the distance of the target, so how do we measure the speed of the target? To measure the speed of the target, the radar sends two chirps separated by "Tc". Each reflected chirp is processed by FFT (Range FFT) in order to detect the range of the target. The range FFT for each chirp will have peaks at the same location, but with different phases. This measured phase difference corresponds to the movement of the detected object with velocity "V".

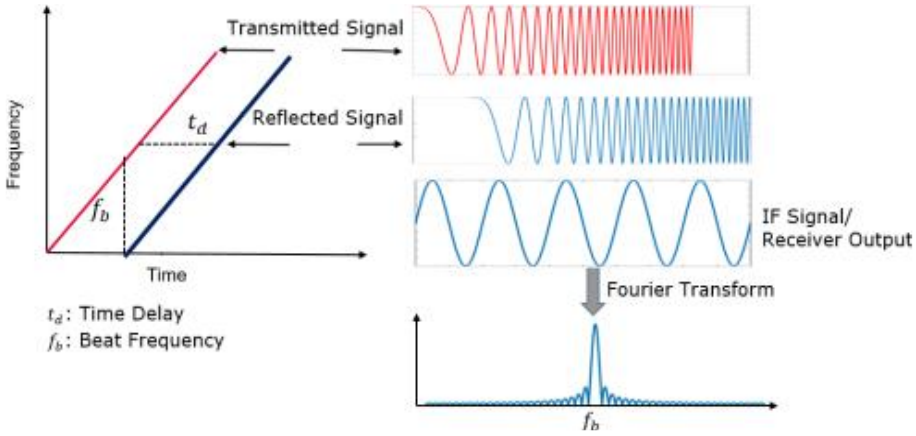

Figure (7). Dual Chirp Velocity Measurement.[97]

Returning to the phase difference issue, the initial phase of the IF signal is the only difference between the phase of the TX chirp and the phase of the RX chirp at the time point corresponding to the start of the IF signal. This time point can be observed on the left dotted line in Figure 6. So, we can get the following phase difference formula [51]:

$$\phi_0 = 2\pi f_c \tau \tag{3}$$

where "fc" stands for the center frequency and "τ" means time delay. Duo to the relationship between center frequency [51], time delay and wavelength. We can transform the function as:

$$\phi_0 = \frac{4\pi d}{\lambda} \tag{4}$$

Where "λ" is wavelength.

Then we can calculate the phase difference between two chirps separated by "Tc", given by:

$$\Delta\phi = \frac{4\pi v Tc}{\lambda} \tag{5}$$

Finally, we can get the velocity of the object, given by:

$$v = \frac{\lambda \Delta\phi}{4\pi Tc} \tag{6}$$

Furthermore, when measuring the velocity of the detected target, it may happen that the phase difference between the difference chirps does not obey the constraint $|\Delta\omega| < \pi$, causing the radar to fail to know the velocity. This happens because the phase difference is a period of 2π. This means that every radar has an upper limit of detectable speed, which means that if $\Delta\omega = \pi$ then the peak manageable speed can be recorded by the following formula:

$$V_{max} = \frac{\lambda}{(4 \cdot T_c)} \tag{7}$$

where the maximum velocity limit can be increased by having closer chirps by decreasing "Tc" and "λ" is wavelength.

Given that the phase shift of millimeter-wave signals is more sensitive to target object movements compared to the beat frequency shift, a velocity FFT is typically carried out across the chirps. This generates the phase shift which is subsequently converted into velocity. The expression for the velocity resolution "ΔVres" can be represented as [51]:

$$\Delta v_{res} = \frac{\lambda}{2T_f} = \frac{\lambda}{2LT_c} \tag{8}$$

In this equation "L" is the number of chirps in one frame, and "Tf" is the frame period.

In addition to distance and speed, another important information is the angle of the target relative to the radar, which is "θ" in the Figure.8 below. Estimating the angle "θ" requires multiple receive antennas. Different distances from targets to multiple receiving antennas will result in differences in the phase of the received signal. The frequency of the received signal basically does not change, because the distance "d" between the receiving antennas is in millimeters, which is negligible compared to the target distance "r". Show in Figure.8.

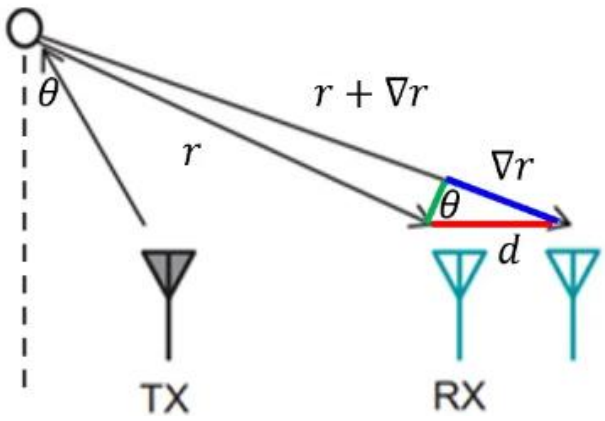

Figure (8). Schematic diagram of target angle estimation

The blue line segment in the above figure represents the distance difference "Δr" between the target and different receiving antennas, and the red line segment represents the distance "d" between the receiving antennas. The relationship between "d" and "Δr" can be expressed as:

$$\Delta r = d \cdot sin(\theta) \tag{9}$$

And ∇r can be expressed by phase difference:

$$\Delta\phi 1 - \Delta\phi 2 = \frac{2\pi\Delta r}{\lambda} \tag{10}$$

According to the above two equations, the formula for angle estimation can be derived:

$$\theta = arcsin\left(\frac{\lambda(\Delta\phi 1 - \Delta\phi 2)}{2\pi d}\right) \tag{11}$$

Similar to velocity estimation, the absolute value of the phase difference also needs to be less than π to ensure no ambiguity, which is, |ΔØ1-ΔØ2|<π. From this, the range of angle measurement can also be deduced, that is, the field of view of the radar:

$$\theta < arcsin\left(\frac{\lambda}{2d}\right) \tag{12}$$

When d=λ/2, the field of view reaches a maximum of ±90 degrees.

To measure the azimuth of a target, at least two receiving antennas are required. When there are multiple targets, it is very difficult for the two receiving antennas to distinguish the targets if they are all at the same range and speed. In order to improve the angular resolution, it is necessary to increase the number of receiving antennas. Let's take a look at how this conclusion was reached.

When there are multiple receiving antennas in Figure.9 the phase difference between each received signal and the previous received signal is ω. Take the following figure as an example, assuming that there are 4 receiving antennas, taking the first receiving antenna as the reference, the phase differences of the 4 received signals are 0, ω, 2ω, 3ω respectively. The change frequency of this sequence signal is ω, so we extract this component through Fourier transform (that is, angle FFT).

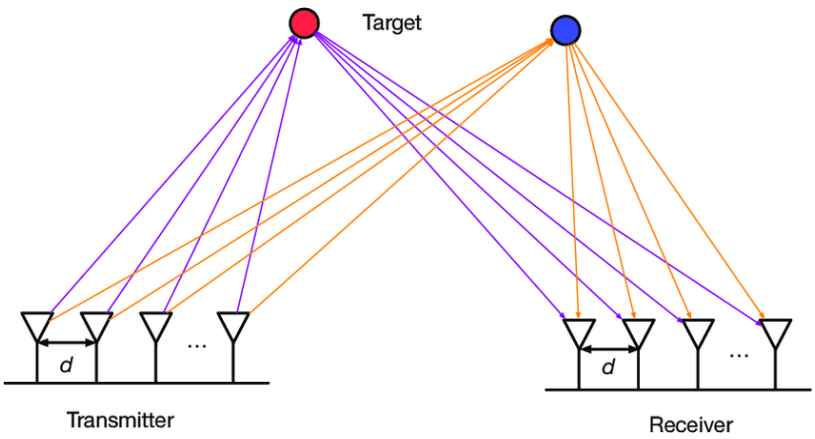

Figure (9). Angle estimation based on multiple receive antennas.

The formula below briefly derives the calculation of the angular resolution. Suppose there are two targets in the scene, the azimuths are "θ" and "θ+Δθ" respectively, and the corresponding phase differences are ω1 and ω2.

$$\omega_1 = \frac{2\pi}{\lambda} d \cdot sin(\theta) \tag{13}$$

$$\omega_2 = \frac{2\pi}{\lambda} d \cdot sin(\theta + \Delta\theta) \tag{14}$$

Since the derivative of sin(θ) is cos(θ), the difference between ω1 and ω2 can finally be written in the form of :

$$\Delta\omega = \frac{2\pi d}{\lambda}\left(cos\left(\theta\right)\Delta\theta\right) \tag{15}$$

According to the Fourier transform theory, the minimum frequency component that can be distinguished by the FFT of point K is 2π/K, where K is the number of receiving antennas. In this way, we can get the smallest angular difference that can be distinguished, which is the angular resolution.

$$\Delta\theta > \frac{\lambda}{Ndcos\theta} \tag{16}$$

"N"means the receiver number, and "d" is the ragne between the receiver antenna.

Generally speaking, take d=λ/2, θ=0, (that is, the center of the radar). At this time, the angular resolution formula is Δθ>2/N. The above formula demonstrates that the angular resolution depends primarily on two factors:

1) *The azimuth angle of the target*. The resolution is highest in the boresight direction. The closer to the edge of the radar FOV, the lower the angular resolution.

2) *The number of antennas*. The angular resolution is directly proportional to the number of antennas. The first factor is beyond our control, and the main means to improve the angular resolution of FMCW radar is to increase the number of antennas. Calculated according to the above formula, the angular resolution that can be achieved by two receiving antennas is about 57 degrees. By analogy, 4, 8, and 16 receiving antennas can achieve angular resolutions of about 28 degrees, 14 degrees, and 7 degrees.

The transmitting antenna sends M Chirp signals in each frame, and the sampling number of each Chirp is N. At the same time, the K receiving antennas will receive K sets of return signals, and the mixer mixes them with the transmit signal to obtain an intermediate frequency signal IF. The IF signal is a three-dimensional data block K x M x N, and the distance, speed and angle of the target can be analyzed by performing three FFT operations on it. The finally obtained RAD data block is the dense underlying data used in the introduction of the millimeter wave radar perception algorithm. Of course, some algorithms also use neural networks instead of FFT. For example, retain the Chirp dimension and use neural networks to extract speed information. Or keep the antenna dimension and use neural network to extract the angle information.

2.3 Radar Signal Processing

As shown in Figure.10, the procedure consists of seven distinct stages. The radar signals (ADC samples) received over a single coherent processing interval (CPI) are initially organized into matrix frames. This results in the formation of a three-dimensional radar cube containing three distinct dimensions: rapid time (represented by chirp index), slow time (illustrated by chirp sampling), and phase (represented by TX/RX antenna pairs).

A 2D-FFT processing technique is then applied to a 3D radar cube to determine the unambiguous range-velocity. Typically, an ADC time-domain signal is subjected to a range FFT to determine the range. Then, in order to determine the relative radial velocity, a second FFT, known as the velocity FFT, is performed across the chirps.

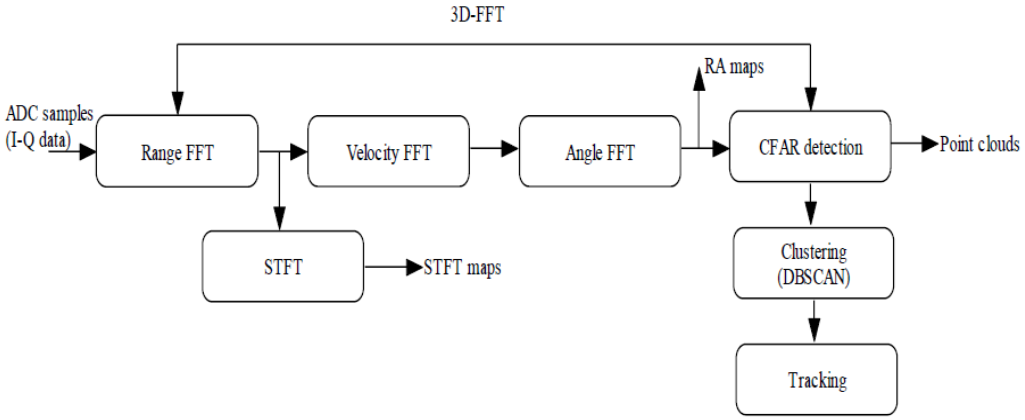

Figure (10). Radar signal processing and imaging. Adapted from [60]

After the conclusion of the first two FFT phases, the entire process yields a 2D map of velocity and range, with areas of greater amplitude indicating potential targets. Nevertheless, distinguishing genuine targets from noise requires additional processing. A Range-Velocity-Azimuth map is generated by performing a third FFT operation on the strongest Doppler peaks within each range segment, known as the Angle FFT. This comprehensive procedure, which consists of the range FFT, velocity FFT, and angle FFT, implements a three-dimensional FFT. In addition, a spectrogram, which is a visual representation of an object's velocity, can be generated by applying a short-time Fourier transform (STFT) to the output of the range FFT.

The fourth phase is targeting detection, which is accomplished primarily by applying CFAR algorithms to FFT outputs. The CFAR detection algorithm is utilized to determine the level of noise in the vicinity of the target, thereby enhancing the precision of target detection. Fin and Johnson introduced the CFAR technique in 1968 [61]; it employs a variable threshold that modifies based on the noise variance in the vicinity of each cell, as opposed to a fixed threshold. Various CFAR algorithms are currently available, each with its own method for

calculating the threshold. In addition, 3D point clouds are generated by applying an angle FFT to the range-velocity bins' CFAR detection. Moreover, this procedure will be utilized frequently in the third chapter.

In addition, a DBSCAN algorithm is utilized to aggregate detected targets into clusters, allowing for the differentiation of multiple targets [64]. Other clustering may somehow supplant DBSCAN in the feature. The final stage of radar signal processing is targeting tracking, where algorithms such as the Kalman filter are used to monitor the position and trajectory of the target in order to provide a more accurate estimation.

2.4 Discuss about FMCW Radar

FMCW radar transmits and receives at the same time, “theoretically there is no ranging blind zone that exists in pulse radar, and the average power of the transmitted signal is equal to the peak power, so only low-power devices are needed, thereby reducing the probability of being intercepted and interfered, short range, distance Doppler coupling and difficult transceiver isolation.” [98]

The performance of FMCW radar to measure the distance and speed of the target has nothing to do with the lighting conditions of the surrounding environment and does not require additional auxiliary light sources to provide illumination. Its higher operating frequency means a smaller overall solution size. “FMCW radar has the advantages of easy implementation, relatively simple structure, small size, light weight and low cost, and has been widely used in civilian/military fields.” [98]

Compared with the pulse radar system, one of the advantages of the FM continuous wave radar is low transmission power, small size, and low cost. The radar can achieve zero blind zone when both the transmitter and receiver are working and can directly measure the Doppler frequency shift. And static target probability, which is very in line with the performance requirements of vehicle radar and industrial radar.

In addition to general indicators, the core performance indicators of this type of radar include resolution, ambiguity, and accuracy of range and radial velocity. The resolution is determined by the signal bandwidth and the coherent processing interval, and the accuracy of parameter estimation is determined by the signal-to-noise ratio of the radar echo signal.

Most contemporary automotive radars now employ the FMCW modulated radar scheme, which has been intensively studied in [50][51]. FMCW radars are rapidly gaining popularity as they are the sensing components of choice in applications such as adaptive cruise control (ACC), autonomous driving, and various industrial uses, including in our current research topic: Human Tracking, Human Activity and Attitude In identification, the most mature and widely used

one is also the FMCW modulation radar scheme. Among them, the most diverse and mature program is launched by Texas Instruments in the United States [65][66], and all the experiments we will discuss later are also undertaken by Texas Instruments equipment.

Chapter III

Points Cloud

The 3D point cloud [35], [36], [37], which is a new representation for objects, is gaining popularity in numerous research disciplines [36] due to its simplicity, adaptability, and potent representation capacity. In contrast to triangle meshes, point clouds do not necessitate the storage or maintenance of polygonal-mesh connectivity [38] or topological consistency [39]. Therefore, processing and manipulating point clouds can result in improved efficacy and reduced overhead. These prominent benefits make processing point cloud research a popular topic.

3.1 Points Cloud Generation

Both range Doppler (or matrix) and point cloud are commonly used data representations in radar applications.

The Range Doppler Map is a 2D representation of radar return data showing the range (range) and velocity (velocity of the Doppler effect) of detected objects. The x-axis usually represents velocity (from Doppler shift) and the y-axis represents distance. These maps are especially useful for radar systems whose main purpose is to determine the range and speed of detected targets. They can provide information about both stationary and moving objects, as shown in Fiugre.12. However, compared to point clouds, range-Doppler maps usually lack precise information about object angles or orientations, which limits their usefulness in applications, but It has to be pointed out that the range Doppler map has outstanding performance in terms of high computational efficiency, which is why most of the micro-Doppler feature data generated by range Doppler were used in the early days. Point cloud data, however, can provide very detailed 3D information about the shape and location of detected objects, which makes them useful for tasks such as 3D modeling, environment mapping, and object recognition. Therefore, the reason why we are more willing to use point cloud data is that computer capabilities have made great progress today.

However, the original signal collected by our millimeter-wave radar is presented in the form of a range-Doppler map, so it is necessary to convert the range-Doppler map into point cloud data. As we mentioned in Chapter 2, the conversion method for converting range-Doppler maps into point cloud data is used extensively in articles applying radar point clouds.

The following is a step-by-step breakdown of the process based on the

flowchart in Figure.10 in Chapter 2, starting with a basic understanding of some of the terms involved:

CFAR detection:
CFAR stands for Constant False Alarm Rate. This is a technique used in radar systems to detect target returns against a background of noise or clutter. It is used to set an adaptive threshold that varies according to the background noise level. Any returned signal above this threshold is considered an object detection.

Angular FFT (Fast Fourier Transform):
FFT is a method of computing the discrete Fourier transform (DFT) or its inverse of a sequence in a computationally efficient manner. The Fourier transform is a mathematical tool that converts A time-domain signal is decomposed into its component frequencies. An "angle FFT" involves performing an FFT along the angle dimension, thereby extracting the frequency components along the angle.

Now, the process of generating a 3D point cloud by applying an angle FFT to the CFAR detection of the range-velocity bins is as follows:

1. First, the radar signal is received and processed to create a range-velocity map.

2. Next, apply CFAR detection to the graph. The CFAR algorithm determines the threshold level for detecting objects in the presence of noise or clutter. If the signal in a bin exceeds this threshold, it is considered detected.

3. Then apply an angle FFT to these detected signals. This FFT is performed along the angular dimension of the range-velocity plot, transforming the detected signal into the frequency domain.

4. The result of this angular FFT can be used to generate a 3D point cloud. Each point in the cloud corresponds to a detected target, and the coordinates of the point represent the distance, velocity and angle of the target.

This method is the most common existing method for translating between range-Doppler signals and point cloud data, which helps to generate a spatial representation of detected objects in the form of 3D point clouds. These 3D point clouds can be used in various applications such as object recognition, tracking, and environment mapping.

3.2 General Processing for Points Cloud

The rapid progress of low-cost sensors such as time-of-flight cameras [48], [43], millimeter-wave radar [33], LiDAR [32] and Kinect [40], [41], [42] has made the acquisition of point cloud data change. The simplicity also enables the rapid advancement of point cloud processing techniques based on these sensor collections. In general, the processing of the obtained human body point cloud data includes the following stages:

1. ***Filtering*****:** Point clouds acquired using these sensors are always subject to noise pollution and contain Anomalies [44,45]. Therefore, filtering operations must be performed on the raw point cloud to obtain an accurate point cloud suitable for further processing. Several techniques can be used to remove outliers and reduce data noise, including statistical outlier removal, pass-through filtering, and voxel grid filtering. This is a crucial stage, as it ensures the quality of the data used in subsequent steps, thereby increasing the overall accuracy and reliability of the system. I looked for some commonly used methods for removing noise from point clouds, which are widely used in various scenarios.

   Statistical Outlier Removal (SOR): This is a popular method for removing noise from point clouds. Computes the average distance of each point from all its neighbors. If that distance exceeds a certain multiple of the standard deviation, the point is considered an outlier and removed.
   Radius Outlier Removal: This method removes points that have less than a certain number of neighbors within a given radius. It is similar to SOR, but uses a radius-based approach instead of a statistical one.

   Voxel Grid Filtering: This method reduces the number of points by building a 3D voxel grid on the input point cloud. Inside each voxel, all points are approximated by their centroids, effectively reducing computational complexity while preserving the overall structure.

   Straight-through filtering: This method is used when we want to focus on a specific region of interest in a point cloud. It allows us to specify a range of acceptable values along a certain axis, and only keep points that fall within that range.

   Conditional Outlier Removal: This is a more advanced filter that removes points based on user-defined criteria. For example, one condition might be that points should only be kept if they lie within a certain distance of a plane or other geometric primitive.

   The choice of these filtering methods above depends on the characteristics of the point cloud and what you want to achieve in the experiment. For example, voxel grid filtering might be a good choice if you want to preserve

the overall shape of objects in a point cloud. But if you are dealing with a lot of noise, you may want to use Statistical Outlier Removal or Radius Outlier Removal. However, according to the conclusions drawn by the author through ablation learning in the article [10], retaining a certain amount of noise will make the trained model have better adaptability and stability.

2. ***Segmentation***: Once the point cloud data is properly preprocessed, it is segmented into different groups. These segments typically represent different objects or parts of objects within the radar's field of view. Techniques such as region growing, Euclidean clustering, or model fitting can be used for this task. The goal is to separate points that might represent people from points that represent other objects or background clutter, making the following steps more manageable and precise.

   As we mentioned in "Radar Signal Processing" in the second chapter, the most common point cloud segmentation method is DBscan. After proper preprocessing, the point cloud data needs to be divided into different groups. These fragments represent different objects or parts of objects within the field of view. This is where DBSCAN comes in. It can be used to cluster points into groups based on their spatial density.

   DBSCAN works by defining a neighborhood around each point. A new cluster is created if at least a minimum number of points (MinPts) (defined by a certain radius epsilon) exist within that neighborhood. This process is repeated until all points have been assigned to a cluster or are considered noise (points without enough neighbors within a radius of epsilon).

   The reason DBSCAN is particularly useful in point cloud segmentation is that because it does not require the number of clusters to be specified a priori, it can find clusters of arbitrary shape, and it has the notion of noise so it can handle outliers. This makes it particularly effective for segmenting complex point cloud data, where the number and shape of objects (and thus clusters) are not known in advance. As show in Figure.11.

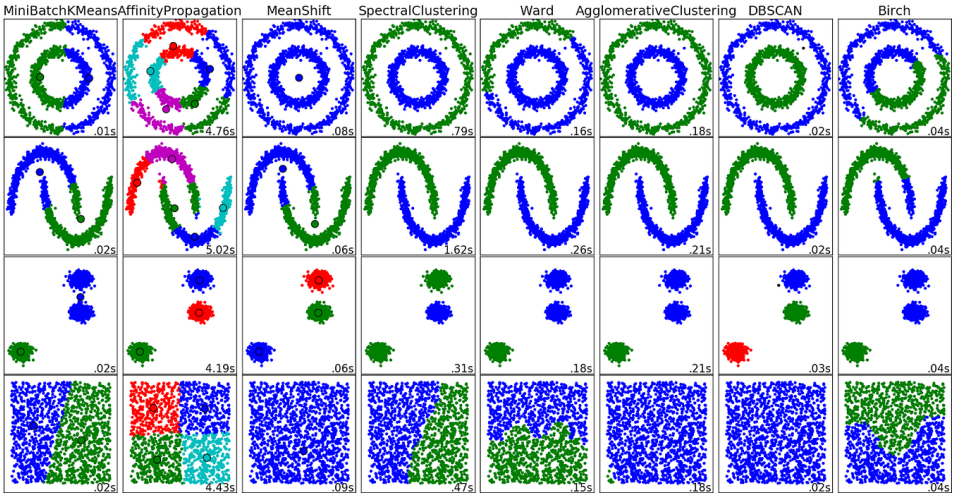

Figure (11). Plot Cluster Comparison

3. ***Feature Extraction***: This phase involves the extraction of certain characteristics or 'features' from each segment of the point cloud. These features can include physical attributes such as height, width, and depth, as well as dynamic properties like velocity and direction. Other potentially useful features could be shaping descriptors, statistical features, or texture features. Advanced techniques like Histogram of Oriented Gradients (HOG) or Convolutional Neural Networks (CNN) might also be utilized. The goal is to extract meaningful and discriminative features that can aid in the identification and tracking of human figures.

4. ***Classification***: The features extracted from each segment are then fed into a classification algorithm, which determines whether the segment is representative of a human. This can be achieved through a variety of machine learning methods, ranging from simpler techniques such as support vector machines (SVM) or decision trees, to more complex deep learning models such as convolutional neural networks (CNN). The performance of this step depends heavily on the quality of the previous steps, especially the extraction of meaningful features. There are plenty of models to try besides SVMs, decision trees, and the very commonly used CNN, let's dig a little deeper into the potential algorithms that can be used for classification:

   Random Forest: Random Forest is an ensemble learning method in which multiple weaker decision trees are combined to form a more robust model. Random forests are less prone to overfitting and generally give better results than single decision trees.

   Gradient Boosting Machine (GBM): A GBM is another ensemble machine learning algorithm that builds multiple weak predictive models, usually decision trees, in a staged fashion. It generalizes to data by allowing the optimization of arbitrary differentiable loss functions.

   Recurrent Neural Network (RNN): An RNN is a type of neural network designed to recognize patterns in sequences of data, such as text, genomes, handwritten or spoken language. This makes them useful for tasks where the order of the data matters, such as time series analysis, language translation, and speech recognition. In the method that will be introduced later, the long short-term memory (LSTM) network is a special kind of recurrent neural network (RNN), which can be used in the classification stage of human activity recognition based on point cloud data. In the works I will introduce [3], [9] have adopted the method based on LSTM model design and achieved very good results.

   LSTMs are particularly well-suited for forecasting on time-series data, or

any data where temporal dynamics are important. In the context of point cloud data, the "temporal" dimension can refer to the sequence of point cloud frames captured over time. As show in Figure.12.

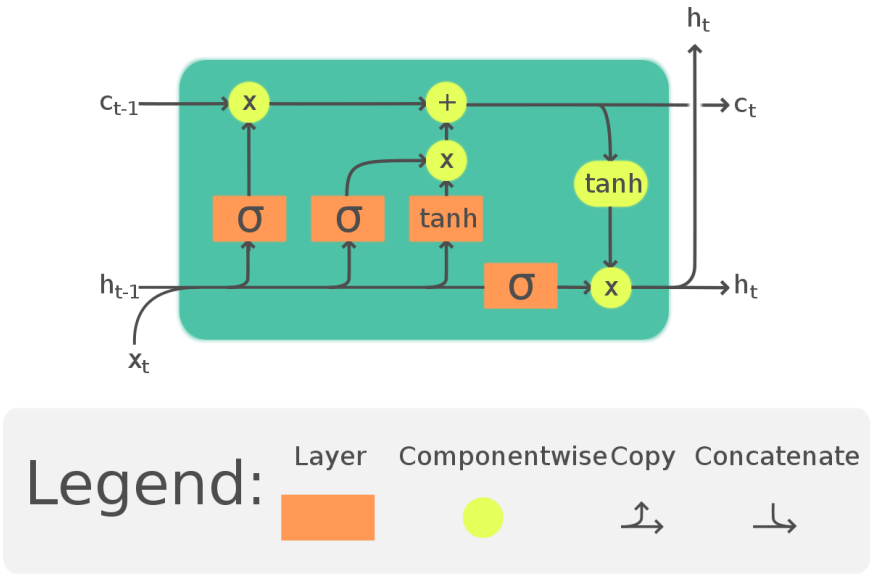

Figure (12). Long Short-Term Memory (LSTM)

Each point cloud frame in the sequence can be processed to extract features such as geometric features, statistical features or deep learning features. These features can then be fed into an LSTM. LSTMs can learn time-series patterns of these features, which are indicative of different types of human activity.

For example, an LSTM can recognize a person walking, running, jumping, etc. using a specific sequence of body poses captured in a series of point cloud frames.

It is worth noting that while LSTMs are powerful, they are also complex models that can require significant computational resources and time to train, especially on large datasets. Furthermore, they require careful design and tuning to prevent problems such as overfitting or vanishing/exploding gradients.

In addition, another model worth noting, Bi-LSTM, is an extension of LSTM, which can improve model performance by presenting sequence data in both forward and backward directions to the model. Essentially, for each time point, the Bi-LSTM layer maintains two hidden layers, one for the forward state and one for the backward state. As show in Figure.13.

In the context of point cloud data for human activity recognition, this means that the model will consider past and future data in the sequence of point cloud frames when making predictions. This can be useful if the current context (activity) depends on what happened before and what will happen

after.

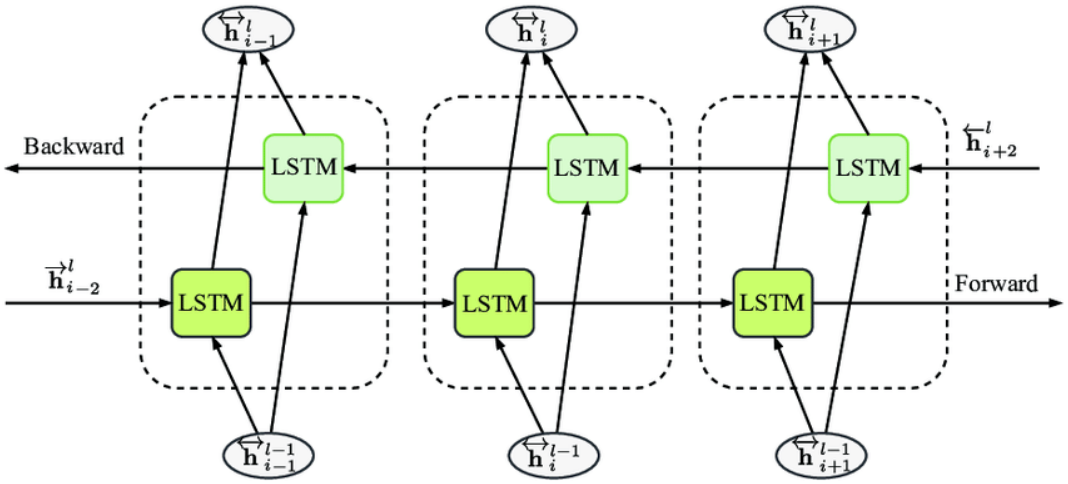

Figure (13). Bi-directional Long Short-Term Memory (Bi-LSTM)

For example, a person may stand up while one is bending over. In this case, seeing the "bend down - stand up" sequence can help the model better understand and classify these types of activities.

Like LSTMs, Bi-LSTMs can be computationally expensive and require careful tuning and training. It is also worth noting that using a Bi-LSTM only makes sense if you have access to the full sequence of frames at prediction time. In real-time scenarios where prediction is made as new frames come in, a standard LSTM may be more appropriate.

Each of these algorithms has its own advantages and disadvantages, and the choice of which algorithm to use will depend on the specific characteristics of the problem and data. It is also important to note that there is no need for a complex model for ordinary classification of human bodies and stationary objects. Here we just introduce the models that will be widely used in this field. For example, LSTM is more suitable for later activities. used in the identification phase.

5. ***Tracking***: If the classification step determines that a human has been detected, the system can track the movement of the person through continuous radar scans. This involves predicting a target's future location based on its previous location, and updating those predictions as new data becomes available. Commonly used tracking algorithms include Kalman filters (assuming a linear motion model) or particle filters (which can handle nonlinear motion models).

   Also, the Hungarian algorithm is widely used here, also known as the Kuhn-Munkres algorithm. In general, the Hungarian algorithm and the Kalman filter are often used together in tracking applications because they each address different aspects of the problem and complement each other well.

   The goal is to find the best assignment that minimizes the total cost, where

the cost is usually some measure of the distance between the predicted position of the object (from the tracking algorithm) and the actual detection in the current frame. In this context, the Hungarian algorithm can solve the allocation problem optimally and efficiently.

The Hungarian algorithm and the Kalman filter are often used together in tracking applications because they each address different aspects of the problem and complement each other well.

A Kalman filter uses a series of measurements observed over time, incorporates statistical noise and other inaccuracies, and produces estimates of unknown variables that tend to be more precise than estimates based on a single measurement alone. Its main advantage is to predict the future position of an object based on its previous position. And high efficiency, making it suitable for real-time tracking. Also, under certain conditions (linear Gaussian models), the Kalman filter provides the best estimate of the system state.

But the limitation is that it assumes that the system is linear, and the noise is Gaussian. These assumptions do not apply to all systems. Also, the Kalman filter can be sensitive to the initial state. But if the initial state is far from the true value, the filter may take a while to converge.

So usually, the Hungarian algorithm is needed to improve the optimization. The Hungarian algorithm is a combinatorial optimization algorithm that solves allocation problems in polynomial time. It is used in tracking to solve the problem of data association, i.e., de terminating which measurements in the current frame correspond to which existing trajectories.

The beauty of the Hungarian algorithm is that it finds a globally optimal solution to the assignment problem, ensuring optimal data association at each step. It is also computationally efficient, especially considering that it finds the global optimum.

In tracking, the Kalman filter is used to predict the future position of an object and the Hungarian algorithm is used to correlate the predicted position with the actual detection in the current frame. Together, they enable tracking of multiple objects over time in an efficient and effective manner.

6. ***Recognition***: After tracking, the next step is recognition, also known as Human Activity Recognition (HAR). At this stage, the specific activity or behavior performed by a person is identified based on the detected movement patterns. Various machine learning and deep learning models can be employed to accomplish this task. Recognition can cover a wide range of activities, from simple actions like walking or standing to more complex behaviors like running, jumping, or even specific gestures. In essence, the "recognition" here is also a classification problem. All the models we mentioned in the previous "classification" stage can be used here. Unlike the previous ones, in order to distinguish between human bodies and stationary objects, the objects we classify here are different. human activities so we will

need more powerful models to perform this task. I won't expand the description here, and this article will use the largest space to give detailed examples of various identification methods.

7. ***Reconstruction***: The final step is human pose reconstruction. The goal here is to reconstruct a 3D model of a human pose from point cloud data. This involves interpreting the data in terms of body parts and their positions. Generally speaking, the most popular method of human pose reconstruction is to identify and locate key points of the skeleton, and then reconstruct the pose of the human body. In this step, the key points of the bones we locate the more accurate the points, the more accurate the human body pose will be when reconstructed. Another key point is the number of key points of the selected bones. Generally speaking, the more key points of the bones, the more accurate the pose we reconstruct, but the increase of bone key points is likely to put higher requirements on the design of the model. The most reconstruction method in the knowledge range uses 25 bone key points, see Table XIII. But this method puts forward very high requirements for the use of hardware. High requirements, and it is not conducive to expanding the use of multi-person scenarios in the future. At the same time, the lack of an open-source database for accurate human body point cloud is also one of the constraints. This step is especially useful for applications that require detailed information about the pose of the human body, such as healthcare monitoring or advanced surveillance systems. Later this article will also use a lot of space to describe this content in detail.

Each of these steps plays a crucial role in the process of human detection, tracking, and activity recognition, and they all build upon each other. The quality and accuracy of the final results heavily depend on the effectiveness of each individual step.

Chapter IV

Human Tracking and Activity Recognition

In the previous section of this work, we outlined the process of using point cloud information. This section of the work aims to describe the latest advances in mmWave radar Human Activity Recognition (HAR).

In recent years, due to the large aging population, the demand for human activity monitoring in public places has gradually increased, and fall detection is performed by tracking the gait of the aging population. The use of traditional sensors such as cameras and lidar has great limitations, such as visual occlusion and obstacles in the blind area, or some scenes are full of thick smoke or water vapor. These traditional sensors can also be affected if the harsh weather in space is turned on. distracted. vision sensor. But it is undeniable that in a good environment, they can detect various activities very well and achieve very high accuracy. These systems represent an effective way to monitor indoor environments, but this raises privacy concerns. At the same time, sensors like lidar also have extremely high costs.

The high sensitivity of mmWave to Doppler-induced frequency shifts makes it suitable for inferring human motion patterns. A widely adopted analysis method is to extract features from the so-called micro-Doppler signature (μD) of the object, which contains time-frequency information about the induced Doppler shift, including the contribution of small-scale motion [5], [6]. μD signatures have been used in the challenging discrimination task of observing subjects from their walking style (gait), which is the aim of the present work.

Human gait has been classified as a soft biometric [7], which means that each individual's gait is unique. However, unlike hard biometrics such as fingerprints or DNA, it cannot be used in high-risk settings or to uniquely identify subjects in very large groups (e.g., more than 1; 000 people). Still, gait is difficult to fake, it can be efficiently analyzed even at a distance, and it doesn't require the cooperation of the subject. For these reasons, mmWave radar-based gait recognition may be a good choice for recognizing objects in scenes, such as surveillance systems or individually customized smart home applications, where the number of people involved is on the order of tens of people, replacing or augmenting traditional camera system.

Chapter V

Using micro–Doppler Characteristics

We noticed that the earliest article [1] on analyzing the micro-Doppler characteristics of millimeter-wave radar to identify human activities in real time came from 2018, published by R. Zhang and S. Cao et al. In this article, the author proposes an innovative method for monitoring human motion behavior. The author collects two kinds of millimeter-wave radar micro-Doppler data. The first is the raw Doppler data collected by the radar without integrated CFAR algorithm. The other is converted from the 3D point cloud information collected from the radar system integrated with the CFAR algorithm an example in Figure.14, and the two kinds of data are used as data sets to train the convolutional neural network model. Finally, the detection and classification of efficient human motion behaviors are realized. Next, I will elaborate on their processing process of using the 3D point cloud data collected in the radar system integrated with the CFAR algorithm.

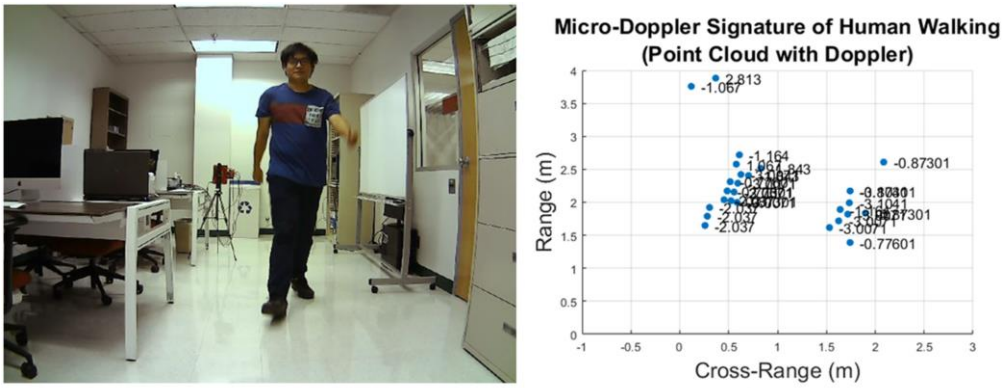

Figure (14). Real-time micro-Doppler prediction scene. (a) Camera vision. (b) Radar point cloud output with Doppler velocities in meter per second [1]

5.1 Real-Time Human Motion Behavior Detection [1]

An example of micro-Doppler characteristics the author uses in the acritical can be seen in Figure.15, where the mmWave radar obtains the micro-Doppler signature of a human walking by raw sampling on board and processing through a host computer. Arm, leg, and torso can be clearly recognized in the entire Doppler data.

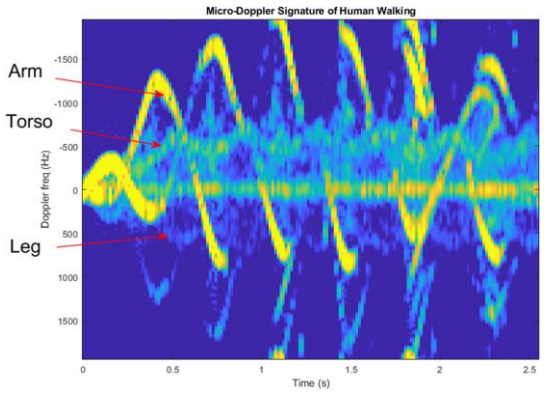

Figure (15). Micro-Doppler signature of a human walking.

5.1.1 Points cloud data generation

Use mmWave radar captures raw range-Doppler data and processes it using the integrated CFAR algorithm to generate point cloud data with Doppler information.

1. *Denoising*: Denoising the raw Doppler radar signals.
   Denoising raw Doppler radar signals typically involves the use of digital signal processing techniques to reduce the amount of noise in the signal and improve its quality. A few of the techniques that are commonly used for this purpose include:
   Wavelet Transform: This is a mathematical technique that transforms a signal into a different representation to make it easier to analyze. Wavelets can efficiently represent and denoise data with transient or spiky behaviors, as well as slowly varying or smooth behaviors. Advantages are it can provide multi-resolution capabilities and can analyze different frequencies of a signal with different resolutions. Also, could preserve high-frequency parts of the signal, which are often lost in other methods. And effective at removing Gaussian noise. Disadvantages are: Requires careful selection of the appropriate wavelet basis function for the specific type of data and noise. But when the performance may be poor if the noise is non-Gaussian.
   Kalman Filter: This is a recursive algorithm that is used to estimate the state of a dynamic system from noisy measurements. The Kalman filter can be used to predict the state of the system in the next time step and update the prediction based on the current measurement.
   Advantages: Effective at handling dynamic systems where the signal is changing over time. Works in real time, as it updates the predictions based on the current measurements. Also, it can estimate missing or noisy data in a sequence. Disadvantages: Assumes that the system is linear and that the noise is Gaussian, which is not always the case. Tuning the parameters of the Kalman filter can be challenging.

Median Filtering: This is a non-linear digital filtering technique, often used to remove noise from an image or signal. It replaces each input pixel value in the signal with the median of its neighboring pixel values. Advantage: Its simple to implement and computationally efficient. And does not require any assumptions about the underlying signal or the noise. Finally, particularly effective at removing "salt and pepper" type noise. Then the limitation maybe, can lead to signal distortion or loss of detail, especially if the window size is too large. And not effective against Gaussian noise.
So, the choice of method should be made based on the specific characteristics of the signal and the noise. In this article the author did not mention which method they use to denoising the raw Doppler radar signals. But we can guess that due to the raw radar Doppler signal has multi-component and non-stationary characteristics, the author may use wavelet transform filtering method here.

2. *Detection*: The integrated CFAR algorithm is employed to detect the presence of objects or targets within the radar data. And the benefits to use CFAR algorithm is that it sets a variable threshold depending on the estimated noise, calculated as the average power level of the neighbors, and filters out points below the threshold or neighbors. This selects only the radar detection points with Doppler information, reducing the amount of data to be transmitted and enabling real-time applications.

3. *Generation of point cloud data*: Once the targets are detected, their respective Doppler shifts are extracted, providing information about velocities and motion behaviors. The detected targets' range, azimuth, and Doppler shift values are combined to create a 3D point cloud representation.

5.1.2 Micro-Doppler signature data generation

The host computer processes the point cloud data using grouping and clustering algorithms (e.g., DBSCAN or other clustering algorithms) to form the micro-Doppler signature data.

1. Time-frequency analysis: The point cloud data is subjected to time-frequency analysis, typically using methods like Short-Time Fourier Transform (STFT) or Wavelet Transform. This analysis helps to identify and isolate the Doppler shifts corresponding to different motion components of the target (e.g., walking, arm swinging, leg movements).
2. Micro-Doppler signature extraction: Based on the time-frequency analysis, the micro-Doppler signature is extracted. The signature is a 2D representation that captures the target's motion characteristics by revealing the frequency modulation patterns associated with the movement. This signature is unique for different types of motion, making it a useful feature for classification tasks.

5.1.3 Convolution Neural Network

In the final recognition stage, the author uses a commonly used convolutional neural network for recognition. The CNN uses leaky ReLU to avoid the "dying ReLU" problem, max pooling to reduce the dimensionality of the feature maps, dropout to avoid overfitting, and fully connected layer to flatten the high-level features learned by the convolutional layers and combine them into the final output. The CNN framework used by the author shows in Figure.16.

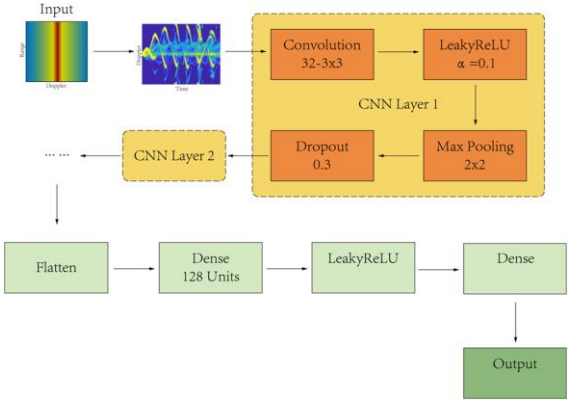

Figure (16). The structure of the CNN network from the radar raw range-Doppler response

In the data collection phase, the author used TI IWR1642 millimeter-wave radar to collect data on a volunteer in the following five human activities: (a) Human walking and vanish; (b) Waving hands when standing or sitting; (c) Sitting to standing and walking transition; (d) walking back and forth; (e) standing and sting still. After the training was completed, the following four human activities were verified. (1) Human walking and vanish from radar got an average accuracy of 96.32%; (2) Human waving hands when standing or sitting got an average accuracy of 99.59% ;(3) Human sitting to standing and walking transition got an average precision of 64%; (4) Human walking back and forth got an average precision of 91.18%. In the case of NO-micro-Doppler detections, 97.84% Average accuracy, in the case of Complex detections including all behaviors, an average accuracy of 95.19% was obtained.

In this result, we can see that this method has a great advantage in detecting moving objects and has very good accuracy. On the contrary, for a stationary human body in a sitting position, the detection accuracy is not high when standing. It can be Said It is relatively low, which is also the biggest limitation of using Doppler data as input data for the machine learning algorithm. Meanwhile, the lower accuracy rate may be due to insufficient training data for transitional behaviors such as standing and walking. These activities occurred for a short period of time, and the authors only used about 1,900 samples for training, as opposed to the nearly 10,000 samples for other activities. Since the applicability of micro-Doppler features goes far beyond the scope of human motor behaviors, such as walking and waving, head shaking, etc., the proposed method can be extended to include other human behaviors.

Chapter VI

Voxelized Points Cloud

We have discussed the early use of micro-Doppler features collected by millimeter-wave radar for Real-time human motion behavior monitoring, and the micro-Doppler features here are converted from 3D point cloud data. The reason why the real-time monitoring method does not choose to use point cloud data is largely due to Micro-Doppler data usually involves fewer dimensions and less data than 3D point cloud data, which can mean less computational resources are required to process and analyze it. This can be particularly important in real-time systems, where there's a need for quick data processing.

However, the relatively low monitoring accuracy of the micro-Doppler feature when the target is stationary has always been a problem that needs to be improved, especially in the activity detection of the target object is the patient and the elderly, the slow action will lead to the monitoring cannot be smooth conduct or even generate false positives.

So we observed that some researchers started to explore the method of directly analyzing the millimeter-wave radar 3D point cloud data for human tracking and recognition [2] or human activity recognition (HAR) [3], [4], [5], [6], [7], [9], [15], [16], [17], within the scope of our knowledge, it was the first to explore and discover the method of directly using 3d radar point cloud signals to detect the human body Activity Recognition (HAR) from the framework proposed by Akash Deep Singh and Sandeep Singh Sandha et al.: ''RadHAR''.

6.1 RadHAR [3]

This is an important work in the field of recognizing human actions or postures using millimeter-wave radar point cloud information. The framework they proposed was cited by many later works and served as the object of comparison.

Below I will introduce RadHAR in detail.

In the early data collection phase of the experiment, they used TI's IWR1443BOOST [65] radar to collect a new point cloud dataset called the MMActvity [3] (millimeter wave activity) dataset. It is an FMCW (Frequency Modulated Continuous Wave) radar that uses a chirp signal. The radar operates in the frequency range 76 GHz to 81 GHz. The radar includes four receiver and three transmitter antennas and can track multiple objects using range and angle information. This antenna design can estimate azimuth and elevation angles, allowing object detection in a 3-D plane [66]. Data from the radar is sent to the laptop via ROS (Robot Operating System) messages on USB. They collected data

on two users. Users perform 5 different activities in front of the radar, as shown in Figure.17. These activities are walking, jumping, jacking jacks, squats and boxing. For subjects performing the same activity, data was collected over approximately 20 seconds in continuity. Some data files are longer than 20 seconds per each. In total, they collected 93 minutes of data for the experiment. Captured point cloud contains spatial coordinates (x,y,z in meters) as well as velocity in meters/second, range in meters (distance from radar point), intensity (decibels) and azimuth (Spend). The sampling rate of the radar is 30 frames per second.

Figure (17): Data collection setup.[3]

6.1.1 3D points cloud processing for RadHAR

1. *Data Splitting*: The authors split the collected data into two separate sets - training set and test set. “They get 12097 samples in training and 3538 samples in testing.” [99]

2. *Voxelization*: In order to solve the problem of uneven number of points in each frame, the point cloud is converted into a voxel representation. Voxels have dimensions 10x32x32 (depth = 10). The value of each voxel represents the number of data points within its boundaries. This process ensures that the input size remains constant regardless of the number of points in the frame. The Voxelization workflow of data preprocessing are show in the Figure.18. The author didn't describe the process of convert, so I think the process of convert should be like this: First, loading point cloud data: Point cloud data can be loaded using software programs that support point cloud processing, such as CloudCompare, Recap, or Point Cloud Library (PCL). Then, Define the voxel grid: The voxel grid needs to be defined. Voxels are 3D pixels that represent physical points in space. The voxel grid defines the resolution and size of the voxels. It is important to choose an appropriate voxel with a size small enough to capture the details of the point cloud data. Finally, assign points to voxels: assign a value to each voxel based on the points that fall within each voxel. This value can be determined by counting the number of

points that fall within each voxel or by computing the average of the points.

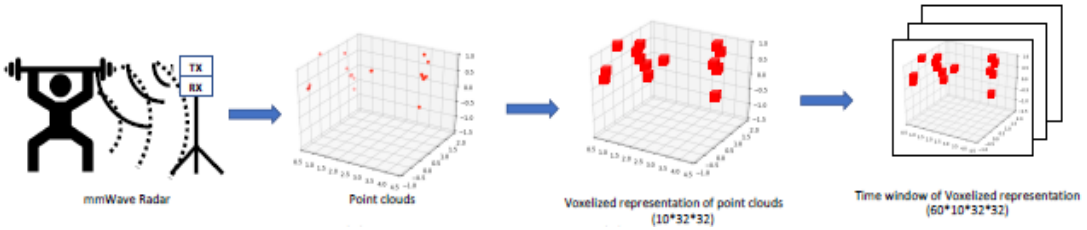

Figure (18): Workflow of data preprocessing. The voxel size if 10*32*32. The time windows are generated by grouping 60 frames (2 second) together.[3]

3. *Time window generation*: To capture the temporal dependence of activity, windows of 2 s (60 frames) were created with a sliding factor of 0.33 s (10 frames). These windows provide a means of analyzing data sequences, taking into account both spatial and temporal aspects of activity. The choice of the 2-second window [67] is based on previous human activity recognition and human recognition studies using point clouds [2].

4. *Classification*: To investigate the results of different classifiers for indoor occupant activity recognition, the authors evaluated several different classifiers on the “MMActivity” dataset. They are support vector machine (SVM) [68], multi-layer perceptron (MLP) [69], Bi-directional LSTM (Bi-LSTM) [74], and convolutional neural network (CNN) combined with LSTM [2]. These classifiers were chosen because they are widely used in various applications, including human activity recognition.

For the result of each classifier, SVM achieved 63.74% accuracy, MLP achieved 80.34% accuracy, and Bi-directional LSTM achieved 88.42% accuracy. The best performing classifier is Temporal Distribution CNN + Bidirectional LSTM with a test accuracy of 90.4 7%. A temporarily distributed CNN layer learns spatial features from the data since point clouds are spatially distributed, while a Bi-directional LSTM layer learns the temporal dependence of active windows. After analyzing the confusion matrix of the time-distributed CNN + Bi-dir sectional LSTM classifier, also can be found that the confusion matrix of one of the trained time-distributed CNN + Bi-directional LSTM classifiers, active jumping and boxing are confused with walking. The reason may be that the data for these activities are similar. The Time-distributed CNN + Bi-directional LSTM classifier trained on velocity voxel representation also had the similar performance.

It is worth noting that the inspiration of LSTM and CNN combined with LSTM architecture comes from another earlier article "mID: Tracking and Identifying People with Millimeter Wave Radar'' [2] that we are concerned about. This article also uses the "sliding time window" " and "voxelization" methods. Next, I will introduce the human body tracking and identification method using millimeter-wave radar proposed by P. Zhao et al.: ''mID''

6.2 mID [2]

The author of this article proposed "a high-precision human body tracking and identification system mID based on millimeter-wave radar" [2]. In addition, according to the author, "This is the first time that the point cloud generated by millimeter-wave radar is used to monitor and identify Research on walking individuals." [2] Therefore, I think the "mID" system has certain originality and certain research reference significance for further research in this neighborhood in the future.

Figure (19): Experiment Setting. "Vicon tracking system" used for collecting ground truth trajectories.[2]

During the experimental setup the authors created their gait recognition pipeline using a commercial mmWave radar IWR1443Boost [66]. The IWR1443Boost [66] millimeter wave radar sensor uses three transmitter antennas and four receiver antennas in terms of hardware configuration to generate and collect 3D point cloud data. "The start and end frequencies are set to 77GHz and 81GHz, respectively, resulting in a bandwidth of 4GHz. The chirp cycle time Tc is 162.14 microseconds, and the frequency slope S is 70GHz/ms." [66] With this configuration, the author can design the "mID" [2] system can detect up to a distance of 5m while maintaining a range resolution of 4.4cm. "It can measure a maximum radial velocity of 2m/s with a velocity resolution of 0.26m/s. The sensor is configured to transmit 33 frames per second, that is, 128 chirps per frame." [66] In order to ensure the accuracy of the experiment The "mID" system was evaluated in a room using the "Vicon Optical Tracking System" to provide the ground truth position of each experimental target within 1 cm. The whole "mID" system consists of radar and backend. As shown in Figure.19, "The radar collects data and generates a 3D point cloud, which is then transferred to the backend computer for further processing. A deep neural network classifier is implemented using the Keras library and the Tensorflow backend." [2]

6.2.1 3D points cloud processing for “mID”

The whole process consists of 4 steps:

i. Point cloud generation
ii. Point cloud clustering
iii. Tracking, using a multi-target tracking algorithm to maintain the trajectories of different people.
iv. Recognition, a recurrent neural network is used to identify user identities from sequential data for each user.

1. *Generation of Point Clouds*: In this article, the authors take a very common practice of using an FMCW radar to transmit mmWave signals and record reflections from all targets in the scene. Then, the designed program calculates the sparse point cloud generated by the object, and then deletes the noisy points generated by the static object.
   In the article, the author does not specify which technique was used to eliminate it, so he presents his own idea. Typically, the process includes the following steps:
   Calibration: The radar system makes initial measurements in a static environment with no moving objects. In order to generate a fixed environment.
   Subtraction: For each new measurement made in the presence of moving objects, the radar system subtracts new point cloud data from the baseline data. Points that do not coincide with a fixed environment are considered "clutter points" and are removed from the dataset. After the subtraction stage, additional filtering techniques, such as clustering algorithms, can be utilized to further refine the data and remove any remaining noise or false positive points. Especially "DBsacn", the algorithm is widely used in point cloud data target recognition and noise removal.
   The main advantage of using "DBsacn" is that there is no need to specify the number of clusters in advance, because individuals enter and exit the monitoring scene randomly. In addition, "DBScan" can automatically identify outliers to combat noise.

2. *Static Clutter Removal*: Points corresponding to static objects, i.e., objects that do not change their position in consecutive frames, are discarded. This can be done by comparing the current point cloud with the previous point cloud. If a point or group of points does not change its position in space from one frame to another, the system can identify it as a static object and remove it from consideration.
3. *Moving Object Tracking*: Track and identify individuals from a continuous point cloud captured by sensors using a combination of detection and association using the Hungarian algorithm and tracking prediction and correction using the Kalman filter. The system essentially creates and maintains a track for object detection at each frame, with inter-frame object

association based on the Hungarian algorithm. If no tracked object is detected for D consecutive frames, the track is marked as inactive and excluded from continuous association. Finally, a Kalman filter is applied to predict and correct the track based on the position and velocity estimates.

The Hungarian algorithm is used to create an association between each object detected and the object kept tracked so that the combined distance loss is minimized, allowing the system to successfully keep detections tracked. The Kalman filter is used to correct for sensor noise and to provide predictive guidance in case a tracked object is not detected due to occlusion or temporary loss of the sensing area. The advantage of this approach is that it can continuously track and identify individuals in real time, even in situations where the object may be temporarily lost or obscured by the sensor's field of view. The use of the Hungarian algorithm and the Kalman filter can help improve the accuracy of the tracking system and reduce false positives and false negatives. However, a potential disadvantage of this approach is that it may require significant computational resources and large amounts of real-time data from multiple sensors. Additionally, the accuracy of tracking systems can be affected by factors such as sensor noise, occlusions, occlusions, and changes in lighting or environmental conditions.

4. *User Identification*: After identifying the points corresponding to human objects, a "tracklet" [71] is used to identify them. Specifically, "they voxelize the points of the intended human subject in each frame of the trajectory using a fixed-size bounding box to form an occupancy grid." [2] But it is worth noting that "OccupancyNet "Grid" itself will contain the subject's body shape information. For example, tall subjects typically have a higher center of gravity, while similarly short subjects have a lower center of gravity.

   The "tracklet" method used in the "mID" system adopts the "sliding window segmentation" method. This method is mostly based on the sampling frequency setting of the system. The author's setting is: "A window consists of 2 seconds of continuous occupancy units, with a 75% overlap with the previous window." [2] Generally speaking, if you extract valuable features directly from the "occupancy grid" to Performing analysis is a very challenging option "because most feature engineering methods are not effective for point cloud classification" [72]. When the "tracklet" identification is completed, the author can identify the "track ID" based on the subject's motion characteristics, such as including the subject's gait and shape information, by providing the classifier with a time-ordered occupancy grid.

   To determine the best ANN architecture for the recognition problem, the authors compared three different LSTM-based architectures [73]. Figure.20 depicts the classification network structure. They adopted a variant based on LSTM because "LSTMs have been shown to be effective for end-to-end

learning on time series data" [2]. In the authors' design: "Each model was trained with the same parameters: 30 iterations and a dropout ratio of 0:5. Each of the three models used LSTM layers with 256 and 128 hidden units. The CNN used in the CNN+LSTM model consists of two convolutional layers and a max pooling layer. The CNN is temporally distributed, which means that the data of each frame is first sent to a two-layer 3D CNN for feature extraction, and then the sequence data is sent to an LSTM for classification." [2]

To evaluate these designs, the authors collected and evaluated a sample of 12 participants. In addition, in order to determine the impact of model size on model classification performance, they also compared various neural network architectures and sizes, and finally they found that bidirectional LSTM networks (with 256 and 128 hidden units) performed best, using the same data In the case of the set, the model achieved an accuracy rate of 89%, and it was found that the smaller group performed better. It can be seen that the size of the model does not determine the quality of the model. Experiments also prove that the recognition method proposed by them has the ability to recognize sparse point cloud data while ensuring high recognition accuracy. However, it is still worth further study that the method may have limitations in the case of large numbers of people or high noise. I think this is also the direction that human body recognition technology needs to intensify in the future, and the accurate identification of target tasks in the case of multiple people.

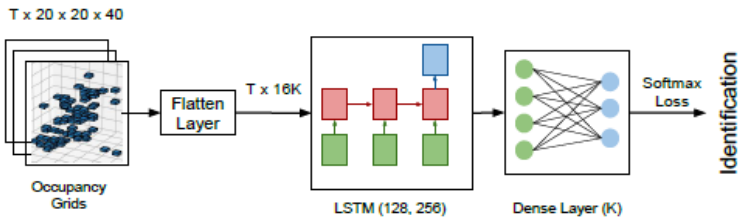

Figure (20). Network Structure. T is the number of frame ; K is the number of people. [2]

Then we can discuss,why does Bi-LSTM work best in this article? I think there are these possible reasons:

1. Temporal Correlations: As mentioned earlier, Bi-directional LSTM processes the input sequence in both forward and reverse directions, enabling it to capture temporal correlations from both ends of the sequence. This is especially useful when working with time series or sequential data where the order of the data points is critical. On the other hand, standard LSTMs only process sequences in the forward direction, which can make it difficult to encode information from the beginning of long sequences.
2. Feature extraction: For the CNN+LSTM model, 3D CNN is used to extract

features from each frame, and then LSTM is used for classification. While a combination of CNNs and LSTMs can help capture spatial and temporal features, the work shows that sparse point cloud data generated by mmWave radars may not be suitable for feature extraction using 3D CNNs. In contrast, bidirectional LSTM can automatically learn more complex features through training, and its ability to bidirectionally process sequences helps capture richer information from data.

3. Convergence: The article shows that Bi-directional LSTM converges faster than the other two architectures, which means it can achieve better results with fewer guesses. The faster convergence can be attributed to the bidirectional processing of sequence data, which enables the network to model temporal dependencies more efficiently.

Overall, the superior performance of bidirectional LSTMs on recognition problems can be attributed to its ability to capture richer temporal correlations in data, automatically learn complex features through training, and faster convergence compared to other architectures.

6.3 Noninvasive Human Activity Recognition

This system consists of four major components: denoising, enhanced voxelization, data augmentation, and dual-view machine learning to lead to accurate and efficient human activity recognition.

The author designed the system use IWR6843ISK-ODS mmWave radar. The complete mmWave radar system includes TX and RX radio frequency components. In our testbed, we use a frequency modulated continuous-wave (FMCW) radar operating at 60–64 GHz with four RX and three TX antennas according to [47] hereof, while the sampling rate is ten frames per second and the instantaneous bandwidth is 400 MHz .The sweep range angle is 120◦ of elevation and azimuth, which leads to the detect ability of movements as small as a fraction of a millimeter.

The point cloud is only sensitive to the moving human body. Their experiments focus on the daily behaviors of elderly people, especially for surveillance of their falls. There are seven types of activities registered in the experiments: 1) walking; 2) changing from standing to sitting; 3) changing from sitting to standing; 4) lying down from sitting; 5) sitting up from lying down; 6) falling down; and 7) recuperating from a fall. To accurately label these activities as the ground truth in their experiments, we have built a synchronous recording platform to integrate radars with a Kinect-V2, which is a depth camera that can record the skeleton's spatial coordinates corresponding to a subject. Using the recorded skeleton's characteristics, such as the skeleton's 3-D coordinates, the angle between the joints, and the displacement difference between the front and

back frames of the joint point, the human behaviors can be labeled (recognized) at the same time.

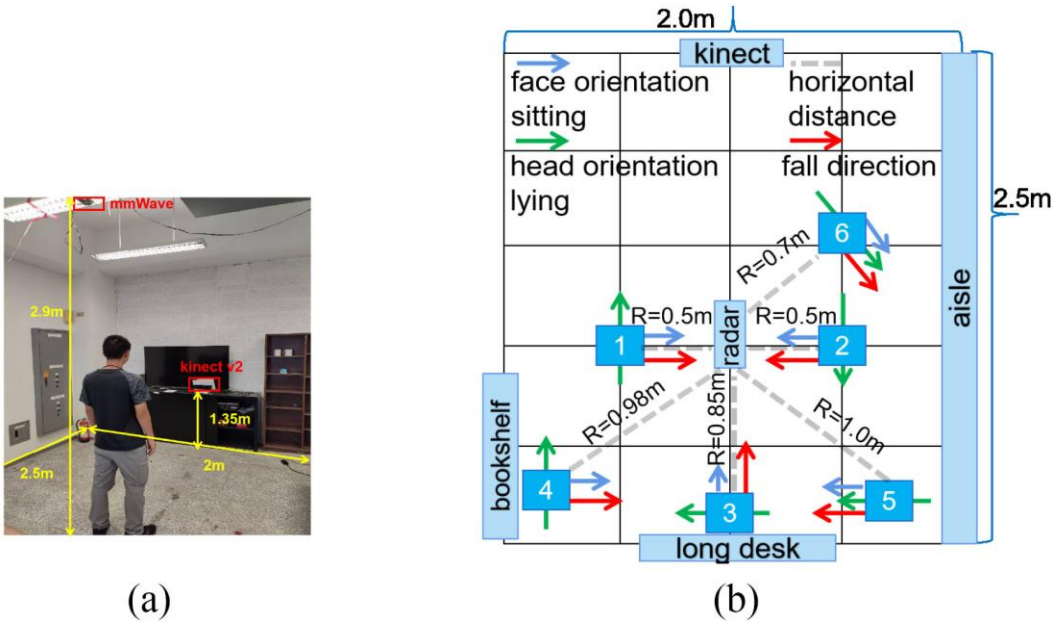

Fig (21). Experimental setup. (a) Lab infrastructure. (b) Floor plan.

The layout of the experiment field is exhibited by Figure. 21(a), the radar was placed at a height of 2.9 m, while a Kinect-v2 camera was placed at a height of 1.35 m to collect the skeleton data in order to label on-going activities. Both radar and Kinect-v2 record the data related to the subject(s) synchronously. Four volunteers (subjects) have participated in their experiments. Their individual heights range from 170 to 183 cm, and their individual weights range from 63 to 85 kg. As shown in Fig. 21(b), these subjects recorded their activities at six different locations in the laboratory. Each activity was performed individually. The experiment is under the assumption that there is only one person in this texting field. The arrows with different colors illustrate the directions of different activities, while the gray-dotted lines mark the horizontal distances from the radar to a subject's locations. The volunteers had stayed in a static condition for 5 s to make our experiments similar to the daily life. Each subject had been recorded for a total of 10 min at each of the six different locations, as illustrated in Fig. 21(b). As a result, we have recorded data for 1200 min since the total duration of each trial in our experiments spans 240 min, while the total number of trials is five.

6.3.1 3D points cloud processing

1. *Denoising*: They use a denoising method based on the DBSCAN algorithm that groups points that are close to each other and marks points in low-density regions as outliers. The benefits are this step helps remove noise from the point clouds and retain only the points that are closely related to human

activity.

2. *Enhanced voxelization*: The authors propose an enhanced voxelization method to transform the point clouds into a fixed-size 3D grid (voxels), which simplifies the processing and analysis.

   Determine the voxel size: The first step is to decide the size of the voxels, which will be used to divide the 3D space into equal-sized cubes. The voxel size should be chosen carefully to maintain a balance between preserving spatial information and reducing computational complexity.

   Quantize the point cloud data: Assign each point in the point cloud to a voxel. To do this, the 3D coordinates of each point are divided by the voxel size, and the resulting values are rounded to the nearest integer. These integer values correspond to the voxel indices in the grid.

   Create a sparse 3D grid: The researchers used a sparse representation for the 3D grid, which only stores the voxels containing points. This helps in reducing memory consumption and computational time.

   Compute voxel features: For each voxel, calculate the local features such as the number of points, average position, and other statistics that capture the essential characteristics of the original point cloud. These features will be used later for human activity recognition.

   Compared with the traditional 'voxelization', the "enhanced voxelization" method has some key modifications: first: instead of creating a dense voxel grid occupying the entire 3D space, the "sparse representation" method is to generate only voxels with points in the point cloud. This can significantly reduce the memory and computational resources required to process voxelized data. The second is "voxel feature computation," a method that not only assigns points to voxels, but also computes additional features for each voxel, such as the number of points it contains, the average position of those points, and other statistics. This process extracts more information from the point cloud, which can improve subsequent analysis and recognition tasks.

   The benefit of these processing steps is that these processes help to preserve the spatial correlation between point clouds in the real environment for further processing and analysis. Different from the traditional voxelization method, which is sensitive to the point distribution, the enhanced voxelization method modifies the boundary of the cubic mask according to the size of the actual test field and the radar sensing range.

Although the raw voxels resulting from the conventional voxelization method, are successfully confined by the identical dimensions for all frames and the

sparsity of the radar point clouds can be mitigated, the relative spatial-correlation information in physical environments would be lost when we focus on the human body point clouds.

This is because the conventional voxelization method was originally designed for objects in a fixed range (hence, the object(s) in the scene would have a constant size on the image plane). The main limitation on the conventional voxelization is that objects' mobility will also be normalized inadvertently. In addition, the cubic mask size is significantly influenced by the distribution of point clouds. When a far isolated point appears, the cube will be enlarged to cover that point. If there are only a few points, the cubic mask will be made smaller thereupon.

In consequence, the conventional voxelization method would be very sensitive to the point distribution. On the contrary, they modify the cubic mask's boundaries based on the size of the testing field and the radar sensing range in reality. They set the dimensions of the cube as 6 m × 6 m × 2.5 m there by, where the radar emitter is located at the ceiling and the maximum human height and the sensing range are 2.5 and 6 m, respectively.

Thus, they apply the same cubic mask for each frame rather than a relative boundary in the conventional method. Therefore, the inter- and intracloud points can convey the human activity and location information, respectively. Figure.22 displays the point clouds captured by a mmWave radar as a person is walking (from left to right). Figure.22(a) and (b) delineates the point clouds generated from the conventional voxelization method and our proposed enhanced voxelization scheme, respectively.

This figure exhibits that it is quite difficult to capture the human moving trajectory based on the conventional voxelization method. On the contrary, the human moving trajectory can be easily observed from the point clouds resulting from our enhanced voxelization scheme. Figure.23 displays the point clouds captured by the radar as a person is sitting down from the standing position. The trend in Figure.23 is similar to that in Figure.22. That is, the underlying dynamics along the Z-axis can be better interpreted by the point clouds resulting from the enhanced voxelization scheme than those resulting from the conventional voxelization method.

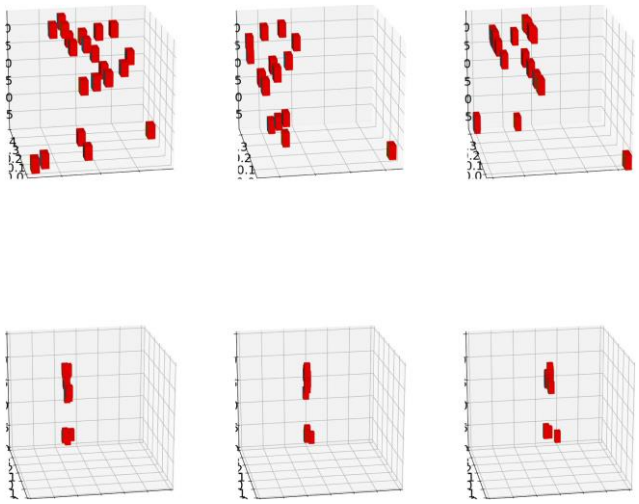

Figure (22). Point clouds corresponding to a person who is walking from left to right. (a) Point clouds generated by the conventional voxelization method. (b) Point clouds generated by our enhanced voxelization method.

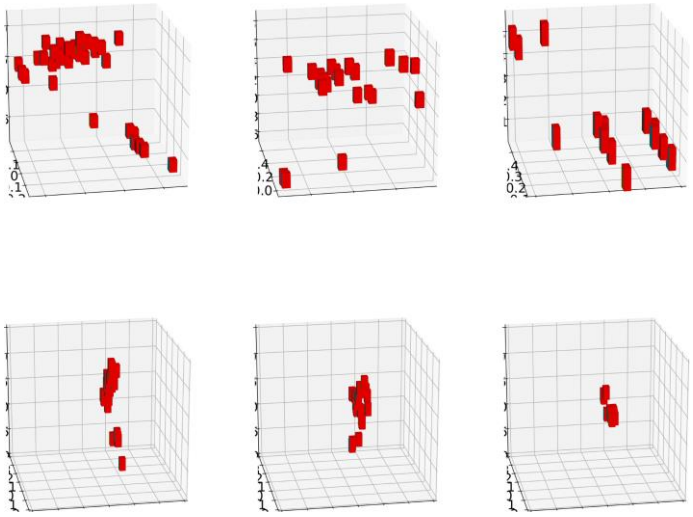

Figure (23). Point clouds corresponding to a person who is sitting down from the standing position. (a) Point clouds generated by the conventional voxelization method. (b) Point clouds generated by our enhanced voxelization method.

3. Data Augmentation: They propose a rotation-shift approach for data augmentation.
    1. Convert Cartesian coordinates (x, y, z) to spherical coordinates (r, θ, φ).

    2. Increment φ by δφ to produce a rotation on the horizontal XY plane. This rotation simulates different angles of view for the radar point clouds.

    3. Convert the rotated spherical coordinates back to Cartesian coordinates using equations (20), (21), and (22). These equations incorporate the rotation (by adding δφ) and shifting (by adding δx and δy) of the point

clouds.

$$x = r \sin(\theta) \cos(\varphi + \delta\varphi) + \delta x \quad (20)$$

$$y = r \sin(\theta) \sin(\varphi + \delta\varphi) + \delta y \quad (21)$$

$$z = r \cos(\theta) \quad (22)$$

4. Perform this rotation-shift approach for a series of angles, each incremented by 30°, effectively creating 12 times more training data than the original data. Figure.24 in the article illustrates this process. The red points represent a person falling at a certain location with the radar placed at the origin. The blue points are the initial group of points rotated 90° clockwise. After shifting the center of the blue point cloud to the origin, the data augmentation simulates a different fall direction using the proposed rotation-shift approach.

   This process can generate various data corresponding to other rotation angles in a similar manner. Benefits: This approach uses the symmetry properties of the radar sensing region to expand the data size by generating rotated and shifted versions of the original point clouds. The data augmentation technique takes the radar as the center of a circle and rotates the acquired point s by a series of angles, making the training data 12 times larger than the originally acquired data.

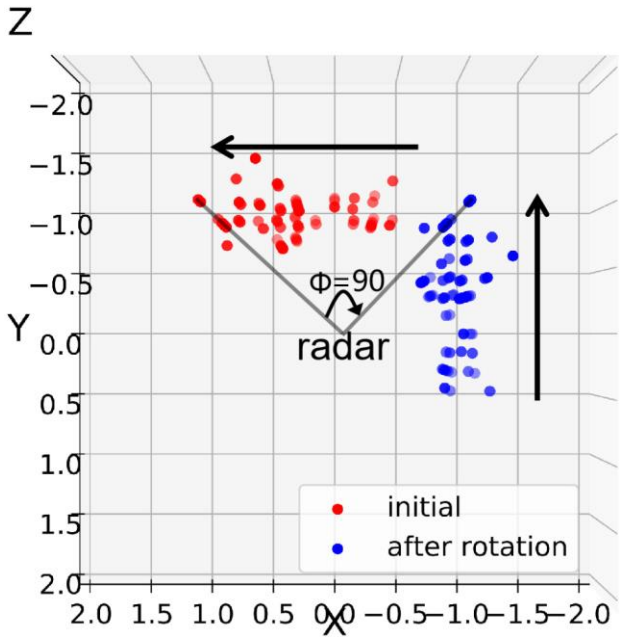

Fig (24). Data augmentation: rotating the initial group of cloud points by 90° clockwise.

4. Dual-View Convolutional Neural Network (DVCNN): The authors propose a Dual-View Convolutional Neural Network (DVCNN) model, the new DVCNN model show in the Figure.25.

   The Dual-View Convolutional Neural Network (DVCNN) can be considered a

kind of forked CNN model. In a forked CNN model, the input data is processed in parallel through different branches of the network, and the results are combined later to obtain the final output.
The authors use the DVCNN model for human activity recognition because it effectively balances accuracy and complexity.

In The last part the author applies a Ablation Study of NN model analysis for various classifiers used in the study, including: SVM, MLP, Bi-LSTM, time-distributed CNN + Bi-LSTM, Dual-View Convolutional Neural Network (DVCNN). Our proposed new method achieves 97.61% accuracy and is more robust, while other existing methods are all below 70% accurate. Among these existing methods, the Ti-CNN+Bi-LSTM method has the best performance with an accuracy rate of 66.79%, followed by the Bi-LSTM scheme with an accuracy rate of 62.53%. The accuracy rates of MLP and SVM are 38.33% and 25%, respectively. The results show that the accuracy of our proposed new method is very consistent on the test data. Other existing classifiers, such as MLP and Bi-LSTM, produce performance outliers.

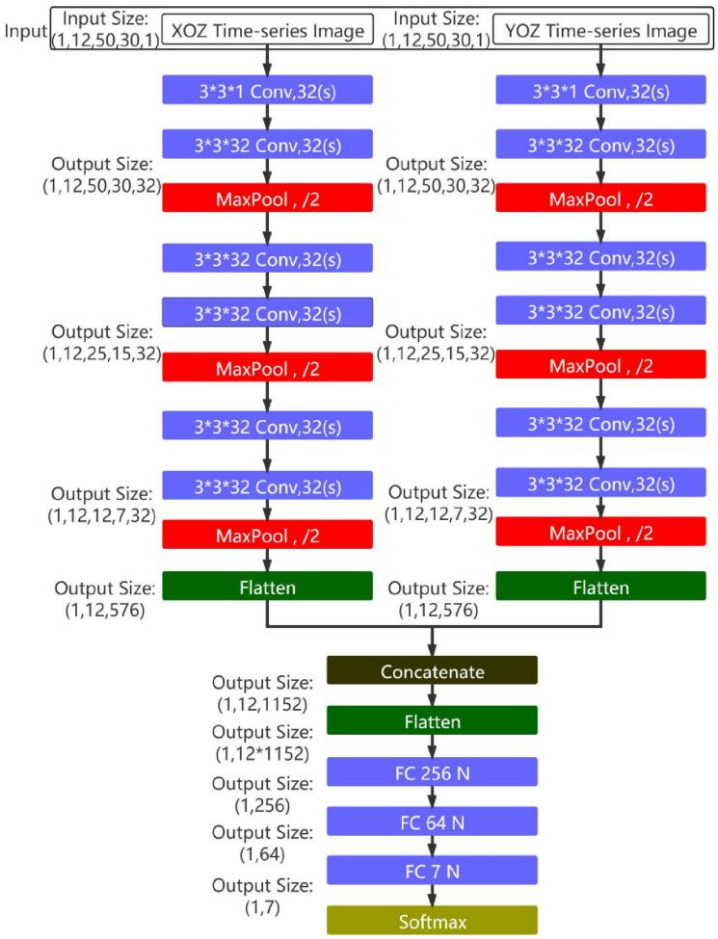

Figure (25). Structure of our proposed new DVCNN.

Second, in the ablation study the authors investigated the influence of various components. Research shows that augmented voxelization methods have the

greatest impact on accuracy (approximately 29% gain), followed by data augmentation (approximately 18% gain) and denoising (approximately 8% gain).

So far,we have shown three articles using voxelized millimeter-wave radar point cloud data in recent years. Their research and application goals correspond to indoor human body tracking and identification, and human activity recognition. In the article, they The data processing methods designed by each have achieved very good accuracy. The experimental goals in the two articles of ''RadHAR"[3] and "Noninvasive Human Activity Recognition Using Millimeter-Wave Radar"[16] are Human Activity Recognition (HAR) in 2019. In the article "RadHAR", the author used the clustering algorithm DBsacn to denoise the point cloud, then voxelized the filtered 3D point cloud, and finally used the fusion deep learning model of Temporal Distribution CNN + Bidirectional LSTM to process the data. Activity classification, for Boxing, Jumping Jacks, Jumping, Squats, Walking and other activities, the processing method finally got an accuracy of 90.47%, and in the article "Noninvasive Human Activity Recognition Using Millimeter-Wave Radar" in 2022 four years later, the author The clustering algorithm is also used to denoise and voxelize the 3D point cloud data. The difference is that the author uses the methods of Sparse Representation and Voxel Features Computation to strengthen the representation of voxels and uses Data Augmentation to make up for the lack of data. The method increases the amount of data, and finally uses a differential CNN model to classify human activities, including: 1) walking; 2) changing from standing to sitting; 3) changing from sitting to standing; 4) lying down from sitting; 5) sitting up from lying down; 6) falling down; and 7) recuperating from a fall. In the recognition of activities such as 97.61% accuracy. From this, I think we can summarize the data preprocessing method of voxelized 3D point cloud and conclude that the voxelization process brings the following advantages and disadvantages to this type of method:

Advantages:

1) The voxelized point cloud data will be stored in the memory in an orderly manner, which is beneficial to reduce random memory access and improve data calculation efficiency.
2) Thanks to the ordered data storage and down-sampling brought by voxelization, this method can handle large-scale point cloud data.
3) Voxelized data can efficiently use spatial convolution, which is conducive to extracting multi-scale and multi-level local feature information.
4) Voxelization is conducive to maintaining the spatial correlation of point clouds, which is convenient for further analysis and processing

Limits:

1) The voxelization process will inevitably lead to information loss, and the degree of information loss is closely related to the selected resolution.
2) The size of the memory footprint and the resolution are almost in a cubic

relationship. When the resolution selection is relatively large, the required memory will be very large.

3) Since the point cloud is spatially sparse, if sparse convolution is not used, a large number of meaningless operations will occur, reducing the operation efficiency.

Chapter VII

Multi-Dimensional Point Clouds

Earlier, we first introduced the direct use of radar micro-Doppler features to recognize human actions in the early stage, and later began to use voxelized [2],[3],[16] 3D point cloud data to train the model to achieve the purpose of Human Activity Recognition (HAR). The data makes it easier to use for model processing to carry more spatial and temporal features. According to this idea, we found that many of the later articles adopted the method of multidimensional data to obtain more information elements in order to directly or indirectly improve the accuracy of the model, but the improvement of the data dimension is bound to be accompanied by model design. The increase in complexity, the increase in training time, the increase in cost and the extension of recognition time. Here I would like to introduce an article in 2019. As far as I know, the method "mm-Pose" [4] proposed by the author of the article, Arindam Sengupta et al. Articles on human activity recognition. Next, I will show the point cloud data processing process in the article in detail.

7.1 mm-Pose

They devised a novel method for real-time human skeleton estimation and tracking using mmWave radar combined with convolutional neural networks (CNNs). Their setup employs a Texas Instruments AWR 1642 step-up mmWave radar transceiver, equipped with two transmit channels and four receive channels on a linear axis. A typical radar in its traditional orientation can only resolve the range (depth) and azimuth reflection point. To circumvent this issue, they use two radars, designated as R-1 and R-2. R-2 is positioned 90 degrees counterclockwise in relation to R-1, thereby transforming the azimuth into the elevation of the reflection point. Each of these radars transmits a wide chirp of 3.072 GHz every 92 microseconds, centered at 79 GHz. To guarantee stable and consistent data acquisition, they utilized a custom-made dual-slot 3D frame for mounting both radars.

The point cloud data processed from both radars were captured via a USB cable, connected to the Robot Operating System (ROS) interface on a Linux computer. Each radar can retrieve up to 256 detection points, including their position (depth, elevation/azimuth), velocity, and intensity within a coherent processing interval at 20 frames per second (fps). Additionally, each data return is accompanied by a header containing a UTC timestamp and radar module index.

For acquiring the ground truth data, they utilized a Microsoft Kinect attached to a Windows computer and employed the MATLAB API. Infrared (IR) sensor data, when coupled with a skeletal tracking algorithm from MathWorks, provided depth, azimuth, and elevation information for 25 joint locations, in addition to UTC timestamps for each frame. A common time server was utilized to synchronize the clocks on the two data capturing computers (radar and Kinect), accepting clock variations on the order of one millisecond. UTC timestamps from both Kinect and radar frames were used to identify and correlate frames.

The experiment was conducted in an open area, with two human subjects of varying sizes, each performing separately. The subjects performed four different actions in sequential sets: normal walking, swinging the right arm, swinging the left arm, swinging both arms. This process resulted in about 32,000 training data samples and around 6,000 samples for the validation/development dataset. They also collected about 1,700 test data samples, where the subjects performed the four actions in a random order to increase the robustness of the experiment.

7.1.1 3D points cloud processing for mm-Pose

1. Collect point cloud: As mentioned in Section 1, radars primarily function as time-of-flight sensors that illuminate a scene using their own RF signal. The phase information of the reflected signal is then used to compute the time delay and estimate the distance of the reflecting point. Due to their millimeter-scale wavelengths, mmWave radar signals are able to detect minute variations in targets. Moreover, radar reflections within the coherent processing interval (CPI) result in a 3-dimensional radar data cube encompassing fast time, slow time, and channels. By utilizing the radar signal processing chain, as outlined in Section III-A, we can extract the range, velocity, and angle information of the reflecting point.

   By applying basic trigonometry, the actual position (x; y; z) of the reflection point relative to the radar (origin) was obtained, where x; y; z represents the depth, azimuth, and elevation coordinates respectively. In this study, the aim was to employ these radar data to estimate the human skeleton using a CNN.

   “There are various ways to represent radar reflection data. The simplest method is a point cloud representation of reflection points in 3-D XYZ space” [4], as shown in Figure.26(a). However, the obvious disadvantage is that this representation method does not provide the size of the reflective surface and lacks a certain indication. So generally speaking, "reflected power level" can be introduced as an additional feature. We define this special parameter as "I". Based on the relationship formula of radar cross section (RCS), we can get: $\sigma = 20log_{10}\left(4\pi R_o^2 A_r\right)$,they assign an RGB weighted pixel value to the points show in Figure.26(b), resulting in a 3-D heatmap, which may serve

as an input to the CNN. Considering the max-range X_{ua}, max-azimuth Y_{ua} and max-elevation Z_{ua} values offered by the radar with resolutions $\Delta x, \Delta y \ and \ \Delta z$, then the input data can be written as:

$$Dimension = \frac{X_{ua}}{\Delta x} \times \frac{Y_{ua}}{\Delta y} \times \frac{Z_{ua}}{\Delta z} \times 3 \tag{23}$$

Let's consider a radar capable of detecting up to 256 reflection points within a single CPI. In order to represent the reflection data within a 5m x 5m x 5m scene and maintaining a resolution of 5 cm across all three dimensions, the input dimensions would equate to a 100 x 100 x 100 pixel image. Each pixel would have three channels (RGB), each corresponding to the intensity of the reflection power. However, two major challenges arise from this approach. First, the CNN would be considerably large and demanding in terms of parameters due to the substantial input size. Second, the input data is extremely sparse (with 256 points scattered across 10^6 pixels), making this an inefficient representation of features and leading to unnecessary computational overhead.

To circumvent these issues, the following method is proposed. Initially, the reflection points are projected onto the depth-azimuth (XY) and depth-elevation (XZ) planes. Following this, a 16 x 16 RGB image is created, where each pixel represents a reflection point, and the RGB channels correspond to the x-coordinate, y/z-coordinate (based on the projection plane), and the normalized reflection power, I, respectively. Pixels that don't correspond to any detections are assigned (0,0,0) across the RGB channels. Consequently, every CPI yields two images, each measuring 16 x 16 x 3. This significantly reduces the input size for the CNN, leading to a substantial decrease in the network's computational complexity.

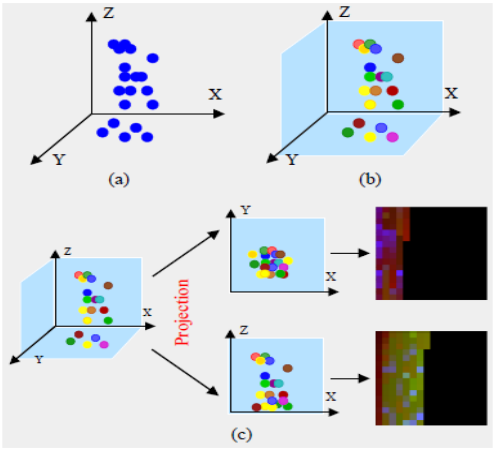

Figure. (26). (a) Point-cloud of the target in 3-D space; (b) Point-cloud of the target with the reflected power use RGB weight represent; (c)Projecting the point-cloud in the XY and XZ planes,[4]

In my opinion, using RGB weighted pixels allows the authors to encode the necessary information for each reflection point in a compact and efficient manner, making it suitable for CNN processing. But it is still possible to use point cloud data without RGB weighting, but it may not be as suitable for CNN as RGB weighted representation, for several reasons:

Information loss: If only point cloud data is used without RGB weighting, the additional information provided by the distance, elevation/azimuth and power level of the reflected signal will be lost. This information is critical for accurate detection and tracking of human skeletons, as it helps the model distinguish points in the point cloud and understand their spatial relationships.

Sparsity: Point cloud data is usually sparse, meaning that there are much fewer points than in dense images. CNNs are designed to work with grid-like data, such as images, and it may not perform well on sparse data without any additional information (such as RGB channels) to help the network learn meaningful features from the data.

Input format: The input format of a CNN is usually a grid of values (such as an image). Directly using point cloud data as input to a CNN may require additional preprocessing steps, such as converting the point cloud to a suitable input format, such as a 3D mesh or depth map. This extra step adds complexity to the overall processing pipeline.

Inefficiency: Due to the sparsity of point cloud data, a large part of the input to CNN will be empty or unused. This results in unnecessary computation and memory usage, reducing network efficiency.

However, other types of neural networks, such as PointNet, which is specially designed for processing point cloud data, may be more suitable for processing raw point cloud data without RGB weighting.

2. Forked CNN and Skeleton Output: After generating the heatmap, the data is projected onto the XY and XZ planes, which is then fed as input to the CNN. The author developed a forked CNN model, depicted in Figure.27, that bears similarities to the DVCNN model referenced in the article "Noninvasive Human Activity Recognition Using Millimeter-Wave Radar" with regard to data input. When it comes to utilizing mmWave radar for human activity recognition, "mm-Pose" is the first work to adopt such a distinctive CNN model. It's noteworthy that due to the small input dimensionality, max pooling layers were not incorporated between the CNN stages. This approach maintained the full resolution of the data without implementing down-sampling operations. The data is then

flattened and processed through a 3-layer MLP (comprising 512, 256, and 128 nodes and using a ReLU activation function) to further refine the non-linear modelling of the input (radar) - output (skeleton) relationship. Lastly, the output layer consists of 75 nodes that represent the (X, Y, Z) coordinates of the 25 joints.

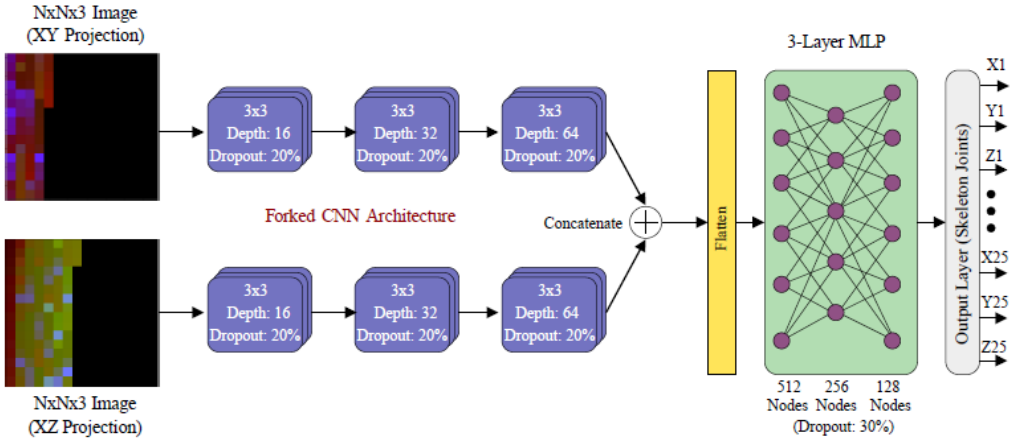

Figure. (27). From the radar projection on the XY and XZ planes, it enters the 3-layer bifurcated CNN architecture, and after being connected and flattened, the output enters the 3-layer MLP, and the final output obtains the spatial position of 25 bone joints. [4]

7.1.2 Results

Upon examining the Mean Absolute Errors (MAEs) across all 25 joint positions, we observed that some indices stood out as outliers during the training phase, yielding the highest MAEs. Furthermore, it was noted that these outliers consistently contributed to high errors across all frames. These outlier joints corresponded to the wrists, palms, fingertips, and thumbs of both the left and right hands. Nevertheless, the remaining 17 joints proved to be effective in reconstructing the human body structure.

In terms of human body reconstruction through bones, the author compared his results with the state of the art, as shown in Table I. When compared to "RF-Pose3D" [75] from MIT Academy, the average positioning error in axis X of 3.2 cm and axis Z of 2.7 cm exhibited improvement, being approximately 24% and 32% better respectively. However, it was found that the positioning error in axis Y of 7.5 cm was higher than the 4.9 cm provided by RF-Pose3D.

Table I: LOCALIZATION ACCURACY COMPARISON [4]

	Localization Accuracy		
	Depth (X)	Elevation (Z)	Azimuth (Y)
RF-Pose3D (MIT)	4.2 cm	4.0 cm	4.9 cm
mm-Pose	3.2 cm	2.7 cm	7.5 cm

7.1.3 Discussion of Differential CNN Model

As mentioned earlier, the differential CNN model is used in both non-invasive human activity recognition and mm-Pose. The authors of this series of articles did not mention why they used the differential CNN model. But I think we can discuss the advantages and disadvantages of the difference CNN model.

First, I think this is due to the size and parameter density of the CNN, since the input size is very large. Second, the input data is extremely sparse (256 points out of 106 pixels), and thus a suboptimal representation of features, resulting in unnecessary computational expense. This is because it can better utilize the spatial information in the 3D point cloud. The bifurcated CNN model consists of two independent CNN branches, one for processing the XY plane and the other for processing the XZ plane. By using a separate branch for each plane, the model can learn specific features and relationships unique to each plane. This can lead to better accuracy and performance than processing the entire 3D point cloud as a single input.

Furthermore, dividing the point cloud into planes also reduces the computational complexity of the network, making it easier to train and run faster. Overall, using a bifurcated CNN model and dividing the point cloud into planes is an effective strategy for processing and analyzing 3D point clouds.
Forked CNN also has certain limitations such as:

Complex architecture:
Since each type of data input requires a separate branch, the design and implementation of a bifurcated CNN model can be complex. This can make it more difficult to optimize and tune the model for best performance.

Overfitting:
Forked CNN models may be more prone to overfitting, especially when one branch of the model is more complex or dominant than the others. This can cause the model to memorize the training data and perform poorly on new or unseen data.

Data dependency:
The bifurcated CNN model relies on different types of data inputs that are relevant and informative for the task at hand. If there is little or no correlation between data inputs, or if one type of data is more important than others, the model may perform poorly.

7.2 Gait Recognition for Co-Existing Multiple People [5]

Earlier we covered the mm-Pose work, which was the first near-by to train a model with multidimensional data. According to the authors of mm-Pose: "mm-Pose can be used in a wide range of applications, including (but not limited to) pedestrian tracking, real-time patient monitoring systems, and through-wall pose estimation for military applications." [4] It can be seen that the future human The applications of activity recognition will be extensive and highly potential. At the same time, in recent years, there is another research direction that is very popular in the field of human activity recognition, that is, "gait detection". According to gait detection, we can realize the tracking, identification or detection of people with special needs, such as patients, the elderly, athletes, etc. Gait recognition has a wide range of potential applications in security checks, health monitoring, and even military.

In terms of gait recognition, people have tried to solve this technical problem in many different ways, such as based on various wireless sensors or based on computer vision or through alternative solutions such as wearable devices. Especially the gait recognition method based on "computer vision" [76]; [77]; [78]; [79] performed well in terms of accuracy, but also has some limitations in terms of privacy protection. First, the camera is used to capture the real Invading people's privacy by using images, which can lead to the disclosure of personal information, especially considering that the camera can be attacked and hijacked. Second, cameras are susceptible to lighting conditions. They cannot capture effective images in low-light conditions. In order to address the aforementioned issues, researchers attempt to acquire locomotion data using wireless signals. The majority of these wireless sensing works rely on channel state information, such as "WiFiU" [80], "wiwho" [80], and "AutoID" [82]. However, WiFi signals are challenging to segment in order to isolate the influence of each individual, making simultaneous identification of multiple individuals impossible.

Next, I present a work by Meng and Z. et al. "Gait Recognition of Coexisting Multiple People Using Millimeter Wave Sensing" [5] also uses the concept of multi-dimensional data, the difference is that different choices are made in the use of models.

First, in the experiment, the author designed two experimental scenarios. They used two millimeter-wave radars from Texas Instruments as experimental data collection equipment to collect gait data of volunteers in two different scenarios, as shown in Figure.28(a). Scenario 1 and Scenario 2. In Scenario 1, the author simulates the situation of "corridor". The field is set as a rectangle. When the subject is close to the radar equipment, because the effective vertical monitoring angle of the radar used is less than ±20°, the equipment cannot scan the subject at this time. The whole body of a person. Therefore, the author sets point M and point N as the reference point and places the two radar devices at a distance of 1 meter from the reference point, and the height of the device is set to 1 m. The first radar uses IWR6843 with the line segment The angle between the site

setting line segment AB is 0°, and the angle between the second radar using IWR1443 and the site setting line segment CD is also 0°. Scene 2 shown in Figure.28(b), the author's simulated experiment the open space is a square, and the living scene that can be referred to is the interior of the office or home. Place two radars IWR1443 and IWR6843 at points E and H, and set the height of the radar to 1m. IWR6843 is diagonal to IWR 1443 and the line segment GH placed." [5]

The authors collected a total of 30 hours of 3D point cloud data from 95 volunteers. This dataset contains two types of walking trajectories: fixed route and free route, in which up to 5 volunteers walk simultaneously.

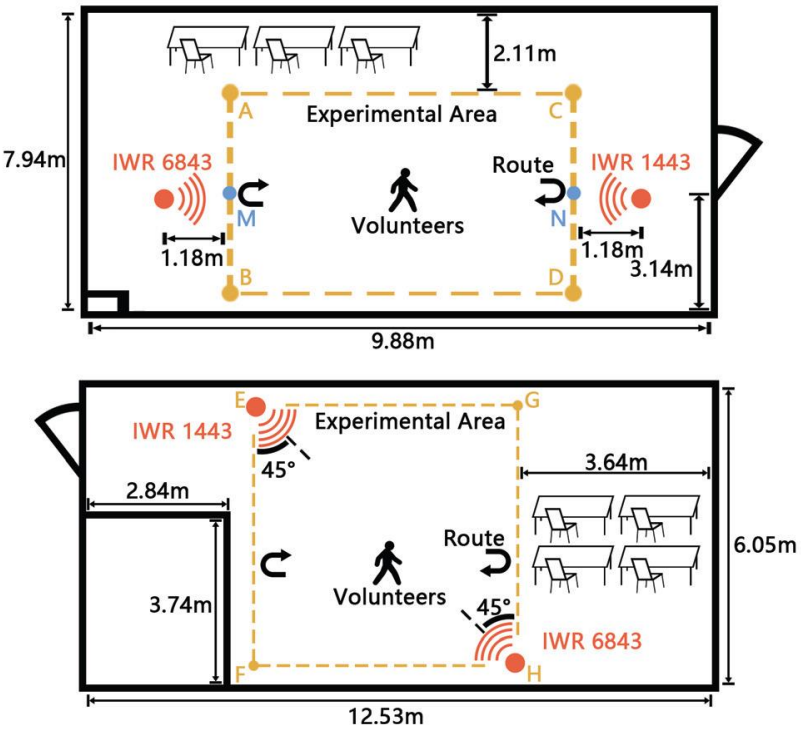

Figure (28): Experimental scenario 1 simulates walking in the corridor facing the equipment. At this time, the effective horizontal detection angle of the equipment is ±60°. Scene2 simulates walking in the living room. At this time, the effective horizontal detection angle of the equipment is ±45°. [5]

7.2.1 Points cloud processing

The point cloud processing method proposed by the author in this article mainly includes the following steps: first, denoise and segment the collected point cloud data, then track and obtain gait data, and then perform data merging. The data is collected by two radars, and finally the model training and gait recognition.

1. *Noise removal and point cloud segmentation*:
   The first step is to use the CFAR (Constant False Alarm Rate) algorithm to remove the noise points reflected by many static objects.
   The second step is to use the DBscan clustering algorithm to remove these noise points and segment the point cloud formed by multiple people.

Here the author adopts the most mature point cloud denoising and segmentation processing method DBscan. The reason for choosing DBSCAN is that it does not need to pre-set the number of clusters, and it can help divide the points in the frame into different groups. Each group represents one person.

2. *Point cloud tracking*: Also in the tracking phase, the author adopted the most mature and effective method in multi-target recognition and tracking then used the Hungarian algorithm to track the clustered point cloud to obtain continuous gait data of volunteers.

   It matches clusters in the current frame with clusters present in previous frames. The weight matrix of the Hungarian algorithm contains the positions of the cluster categories generated in the last 10 frames, which helps to alleviate the cluster interruption caused by the sparse point cloud data.

3. *Data Merging*: In this step we will ask why do they need to merge data? The reason is that they decided to use two devices to collect gait data at the same time, which can greatly increase the number of points in the point cloud and reduce the mutual coverage of volunteers.

   Data merge process:

   The first is coordinate transformation: through the rotation and translation of the coordinate system, the point cloud data is converted into the same coordinate system. First, rotate the coordinate systems of both devices clockwise to align their orientations.

   Next, using the translation formula involving the rotation angle of the coordinate system and the coordinates of each point in the original coordinate system and the new coordinate system, the coordinate system of IWR6843 is translated to be consistent with the coordinate system of IWR1443.

   The second step is the merging process: the point clouds collected by the two devices are merged according to their timestamps. Each point cloud is given a new attribute called device name, which records the ID of the device that collected the point cloud. The coordinate-transformed point clouds collected by the two devices are merged into the same file, and all point clouds are sorted according to the acquisition time. The point clouds of two devices whose time difference is less than the specified threshold (set to 50 milliseconds) will be merged. The average value of the time difference for merging point clouds was found to be 24ms.

4. *Millimeter Gait Network* (mmGaitNet): This neural network model is specifically designed to process point cloud data for gait recognition. The input data represents human motion in the form of a 3D point cloud. The three-dimensional coordinates of the point are represented by X, Y, and Z, V represents the radial velocity, and S represents the signal strength of the

point.

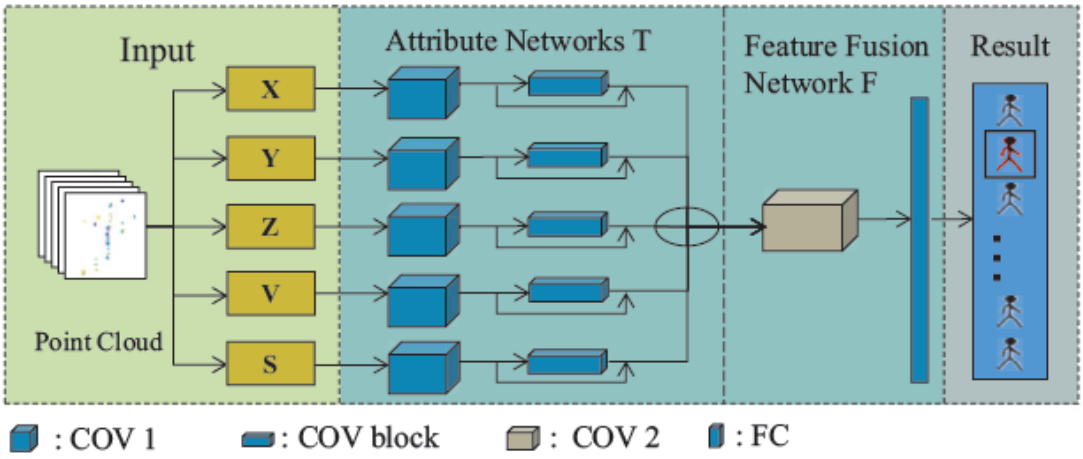

Figure (29): mmGaitNet framework structure. The 7×7 spatiotemporal convolution kernel with 2×2 stride is used in Attribute Networks named COV1; the layer1 of ResNet18 is represented by COV block also in Attribute Networks; a 3×3 spatiotemporal convolution kernel with 1×1 stride Layers are denoted by COV 2; fully connected layers are denoted by FC. [5]

The structure of the model Figure.29 is designed to handle these five properties efficiently. The mmGaitNet model consists of five identical attribute networks and one fusion network. Each attribute network takes as input a point cloud of a single attribute, represented as a p x t matrix, where "p" is the number of points in the point cloud and "t" is time. The attribute network extracts feature information for each attribute separately. The fusion network then combines the features extracted by each attribute network to create an overall representation of the point cloud. It then goes through a final fully connected layer to output the class score, which represents the identified person.

5. *Robustness verification of mmGaitNet*: They conducted experiments on mmGitNet in two different scenarios, and the results showed that the change of the scene had no effect on the recognition accuracy. They evaluate our method on data collected when two people walk simultaneously in two scenarios. In Scenario 1, our method achieves 86% accuracy. In Scenario 2, our method achieves 93% accuracy. Experiments validate the performance of mmGaitNet to adapt to environmental heterogeneity. The reason is that mmWave sensors and our signal processing algorithms are able to eliminate static reflection points from different furniture in different environments.

   So, I think we may have a question that why is the effect of scene 1 better than scene 2?
   The reason why the result of Scenario 1 is better than that of Scenario 2 is

because of the location of the radar sensor. Due to the limited launch angle of the radar wave, the radar horizontal detection used in this work is 45° and 60° respectively, so the radar sensor similar to Scenario 2 is placed diagonally Can achieve better test results.

7.2.2 Results and Discuss

Compared with the mm-Pose in the previous work that uses the RCS feature of the radar to add RGB weights to each point cloud to form a heat map, the author here uses each original Doppler radar signal. Species characteristics, three-dimensional coordinates, speed, signal strength. An attribute network is used to take each feature as a separate input. This greatly reduces the difficulty and speed of processing fusion data by a single network, and finally outputs the classification result after fusion of the processed data. This method of operation is very similar to the idea of differential CNN but executed more thoroughly. Each function is handled individually.

Finally, the author also compares many existing models.

First, the author's gait recognition accuracy for each person using the IWR6843 sensor. This table compares the performance of four methods: PL (PointNet with T-Net and LSTM), P-L (PointNet without T-Net and LSTM), DR (Deep Residual Neural Network) and the proposed mmGaitNet. The results in this table show that mmGaitNet consistently outperforms other methods for every person and every scene. Regardless of the number of volunteers walking simultaneously, mmGaitNet outperforms PL, P-L, and DR in accuracy. Furthermore, the table shows that the gait recognition accuracy of all methods generally decreases as the number of volunteers increases. However, mmGaitNet can maintain relatively high accuracy even in more complex scenarios where multiple people walk at the same time.

Second, the authors use two sensors (IWR6843 and IWR1443) for the accuracy of gait recognition for each person. Still the same four methods: PL (PointNet with T-Net and LSTM) [83], P-L (PointNet without T-Net and LSTM), DR (Deep Residual Neural Network), and the proposed mmGaitNet (ours). Likewise, the results show that mmGaitNet consistently outperforms other methods for every person and every scene when using two sensors. And regardless of the number of volunteers walking simultaneously, the accuracy of mmGaitNet is higher than PL, P-L and DR.

These two comparative experiments clearly show that the gait recognition accuracy of all methods improves when two sensors are used instead of only one (IWR6843). This improvement was especially pronounced when multiple people were walking at the same time, suggesting that the performance of gait recognition systems can be enhanced with the use of additional sensors.

Then the author also discussed the impact of each feature on the collation accuracy. Gait recognition accuracy of the mmGaitNet method when removing different attributes from point cloud data. This experiment compares the

performance of the method under six different conditions: removal of X coordinate (no_X), removal of Y coordinate (no_Y), removal of Z coordinate (no_Z), removal of radial velocity (no_V), removal of signal-to-noise ratio (no_S), and use all properties (Ours). The results in this table show that removing a single attribute leads to a drop in accuracy compared to a method that uses all attributes (ours). When all attributes are included, the method is most accurate at 90%. Table II show the results.

Removing each individual attribute affects the accuracy of the method differently. Removing the X coordinate (no_X) was 77% accurate, while removing the Y (no_Y) or Z (no_Z) coordinate was 83% accurate. This suggests that the X coordinate may have a more important role in gait recognition than the Y and Z coordinates. The accuracy of removing radial velocity (no_V) and signal-to-noise ratio (no_S) was 82% and 86%, respectively, suggesting that these properties also contribute to the overall performance of the method. This shows that each attribute contributes to the effectiveness of the method, and the combination of all attributes provides the most accurate and robust gait recognition.

Table II: Accuracy of each person under different attributes.

Method	noX	noY	noZ	noV	noS	Ours
Accuracy	77%	83%	83%	82%	86%	**90%**

At the end of the work, the authors emphasize the importance of considering different input formats of point cloud data when designing gait recognition algorithms. The results show that the similarity between coordinates, radial velocity, and signal-to-noise ratio is smaller than that between individual coordinates. Furthermore, the attributes are independent of each other and represent different features of the point cloud. Second, the way point cloud attributes are processed and combined plays a crucial role in achieving the best gait recognition accuracy. The proposed mmGaitNet method, using a separate network for each attribute, outperforms other methods, demonstrating its effectiveness in processing point cloud data for gait recognition tasks.

7.3 Human Gait Recognition Using Multi-Channel-3D-CNN [7]

This is also an article about gait recognition in the field of human activity recognition. The author of this work: Jiang, X.; Zhang Yong and others proposed "a human gait classification and recognition method based on millimeter wave array radar". The author proposes a multi-channel 3D convolutional neural network, which achieves the purpose of human gait recognition through the extraction and fusion of multi-dimensional features of information collected by millimeter-wave radar, which is also a typical classification problem. "[7].

For radar placement and data collection the author designed the system in such a way that the IWR1443BOOST [66] radar system was placed at a height of 0.8 m above the ground. The radar system design in Figure.30. The target (person) is allowed to move along a non-fixed path within the detection range of the radar. Subjects, in turn, performed various types of locomotion in the experimental field, including normal walking, jogging, limping (with one leg trailing behind), squatting, and standing up. The movements are performed in different scenarios, including corridors, basketball courts, and parking lots, to increase the diversity of the sample data.

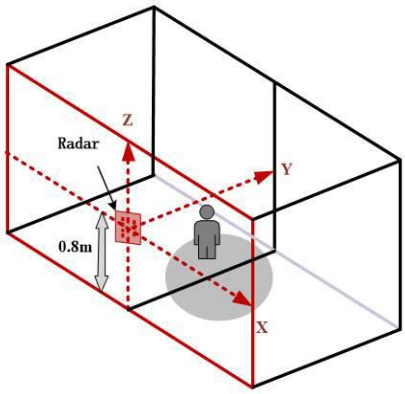

Figure (30). Experimental scenario: Schematic diagram of experimental scene.

Overall, the first step is Generation of a point cloud of human gait. The FMCW radar is capable of measuring the range, velocity, and angle of a target by transmitting FMCW signals. There are three main processes covered in this step:

1) Range Measurement
2) Velocity Estimation
3) Direction of Arrival (DOA) Estimation

Once the above information is known for every detected point on the target, a point cloud can be generated. Each point in the cloud would represent a point on the target, with the position of the point in the cloud defined by the calculated X, Y, Z coordinates.

7.3.1 Points cloud processing

The whole 3D points cloud pre-processing has 5 steps, which are:

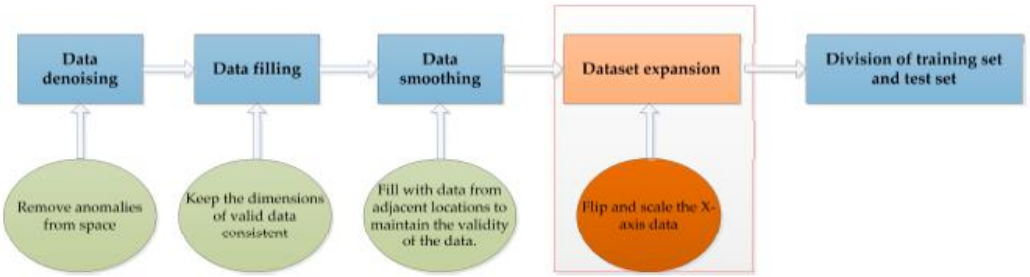

Figure (31). Flow of data processing.

1. *Data Denoising*: During this stage, a distance threshold is set in the 3D space to eliminate noise points or outliers from the data. This ensures the data going into the 3D spatial coordinates, radial velocity, and intensity channels of the neural network is as clean and accurate as possible.
2. *Data Filling*: This stage is to make sure that each frame of data contains 64 points cloud, thereby maintaining the consistency of the input data dimensions (64 points cloud per frame). This is done by filling in null values in frames with fewer points. After data filling, each of the 3D spatial coordinates, radial velocity, and intensity channels will have an input size of 64 * 40 for each training sample.
3. *Data Smoothing*: After dimension expansion (data filling), the point cloud data can't be null because null values can't provide useful data for the convolution process. So, the data of adjacent points is copied to create a 5D array containing 64 valid points for each frame of data. This means each of the three channels (3D spatial coordinates, radial velocity, and intensity) will have a well-structured, non-null input for the neural network.
4. *Dataset Expansion*: If the Sample Size is too Small, it is needed to expand the dataset to enhance the robustness of the neural network mode. Flipping and Cropping May Be used for this Purpose. This affects all Three Channels of the Neural Network input, increasing the amount of data they each must work with.
5. *Division of the Training Set and Test Set*: Here the author used N-fold Cross-validation. In order to evaluate the network model, they used an N-fold cross-validation strategy. Here, N = 4, meaning the sample data was divided into four equal parts for four-fold cross-validation (CV).
   After the dataset is fully prepared, it is divided into a training set and a test set in Table III. The training set is used to train the neural network, and the test set is used to evaluate its performance. Each of the three channels (3D spatial coordinates, radial velocity, and intensity) will have their data split into a training set and a test set.

Table III. Cross-training and dataset partition.

Cross-Validation	Dataset 1	Dataset 2	Dataset 3	Dataset 4
CV_1	Training	Training	Training	Testing
CV_2	Training	Training	Testing	Training
CV_3	Training	Testing	Training	Training
CV_4	Testing	Training	Training	Training

6. *MC-3DCNN classification*: In this work, the authors design a multi-channel 3D convolutional neural network (MC-3DCNN) for the classification goal of human gait. This type of network is a neural network specifically designed for analyzing and understanding features in multidimensional data. This

design enables the model to independently extract features from multiple types of input data. At the same time, it has many similar ideas with the differential CNN and multi-attribute network we introduced earlier, especially the internal structure is very similar to the DVCNN model mentioned above, the main difference comes from the different input structure of the data. The structure of the Neural Network model show in the Figure.32.

The motivation behind the design of the MC-3DCNN model is based on the principle of feature fusion and augmentation. The network structure is designed to independently process and extract features from the radar signal on three different types of data (3D spatial coordinates, radial velocity, and intensity) representing different physical properties of the detected motion. By processing these channels independently, the model can gain a more complete understanding of the actions performed by the human body.

These three data channels are not merged until later in the model architecture. The advantage of this multi-channel design is that it allows the network to learn different feature representations before fusing each type of data together. This could allow the model to capture the finer nuances and complexities in motion, leading to better motion recognition performance.

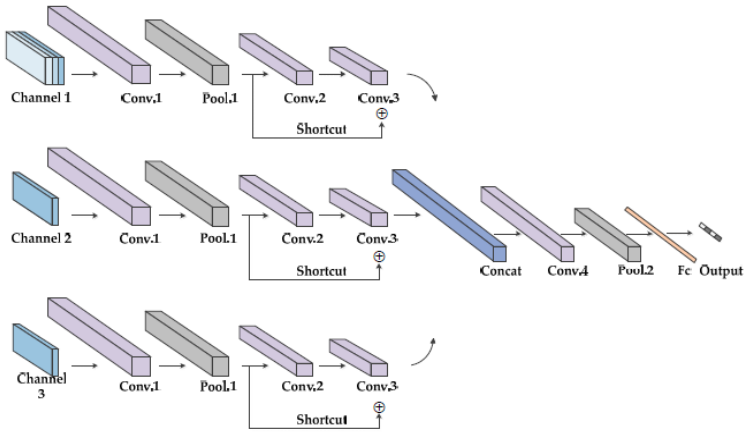

Figure (32). Structure of multi-channel three-dimensional convolution neural network (MC-3DCNN).[7]

The structure of the NN input:

In this work, they uniformly segment gait sequence samples, and each gait segment is set to last 2 seconds, covering a complete gait cycle. Given that the radar is sampled at 20 frames per second, this would result in 40 frames (2 seconds) of gait data as training samples. They split this data into three channels: 3D spatial coordinates, radial velocity, and intensity, which are the input to a multi-channel 3D convolutional neural network (MC-3DCNN).

3D space coordinates: These are the X, Y, and Z coordinates of each point in the point cloud data. This data represents the position of each point in

three-dimensional space. The input size of this channel is 3 * 64 * 40.

Radial Velocity: This is the velocity and direction in which each point is moving directly away from or towards the radar. This velocity is expressed as a scalar (one-dimensional). The input size of this channel is 1 * 64 * 40.

Strength: This data represents the power or strength of the radar signal returned by each point. Like radial velocity, intensity is a scalar quantity. The input size of this channel is 1 * 64 * 40.

We can think about the following why they chose to preprocess the data in this way?
Personally, I believe that the reason they chose to preprocess the data in this way, in the context of this study, may have to do with the need for a structured and consistent format to feed into the neural network model. By segmenting the gait data into 2-second periods, they ensured a consistent input size for the model. By separating this data into three channels (3-D spatial coordinates, radial velocity, and intensity), they were able to independently analyze and learn the unique characteristics provided by each data type, which can improve the model's ability to recognize and classify different human's gait.

7.3.2 Results of the classification

Table VI. shows the recognition results of the cross-validation. From the identification results of cross-validation, it can be seen that MC-3DCNN can well identify the movement with strong continuity with an accuracy rate of more than 90%, but for Lame with weak continuity and less obvious micro-Doppler features the accuracy of walking recognition is relatively low. So, this is also one of the development directions of future work, to improve the recognition accuracy of actions with weak continuity and less obvious micro-Doppler features.

Table VI. Recognition results of cross validation [7]

Accuracy (%) / Category	CV_1	CV_2	CV_3	CV_4	Average Accuracy
Jogging	95.20	89.80	94.60	90.40	92.50
Normal walking	90.40	95.20	96.80	89.60	93.00
Lame walking	81.60	94.20	89.60	85.60	87.75
Squatting down and standing up	94.80	92.50	89.80	93.60	92.68

Of course, in order to prove that the proposed "MC-3DCNN" [7] "achieves higher recognition accuracy by enhancing and fusing the features of spatial coordinates, radial velocity and intensity in the three channels, rather than just increasing the training samples quantity , So as to the Achieve Good Results."[7] They designed "an experimental Single-Channel 3D Convolutional Neural Network (SC-3DCNN). The Structural Design of SC-3DCN N is similah to mc-3dcnn, but it integrates all

three features into one channel for ele each of the 40 frames of data."[7]
The recognition results of different networks are shown in Table V.

Table V. Recognition results for different networks.[7]

Accuracy (%) / Network	CV_1	CV_2	CV_3	CV_4	Average Accuracy
SC-3DCNN	84.20	89.80	81.60	89.60	86.30
MC-3DCNN	90.50	92.93	92.70	89.80	93.00

It can be seen that the Muti-Channel model can better recognize the three types of motion with strong motion continuity than the SC model, but for the gait categories with weak motion continuity and less obvious micro-Doppler features, the recognition accuracy is almost the same, We can thus conclude that while MC-3DCNN has clear advantages in recognizing certain types of strong continuous motion, there is still room for improvement in recognizing irregular or less dynamic motion. Further research or refinement may help improve.

7.4 MARS [10]

Several articles have been presented that use multidimensional point cloud data to analyze human activities or locomotion. Similar to “mm-Pose”, the output of the neural network model in the final stage of human posture recognition is not directly proportional to the pose. The result of classifying the human body posture is the spatial position information of the 25 joints, after which the human body posture is restored via bone reconstruction. Figure.33 depicts an example of angle estimation by "MARS" [10] during squatting movements as reconstructing skeletal information to recognize human posture has progressively garnered traction in recent years. For instance, rehabilitation is a crucial procedure for some elderly individuals and patients with movement disorders. Rehabilitation exercises are currently performed under the supervision of clinical experts. To enable patients to perform prescribed exercise at home and reduce commuting requirements, specialist shortages, and healthcare costs, novel approaches are required. The estimation of human joints is an integral part of these programs because it provides valuable visualization and feedback based on body motion. Popular camera-based systems are used to capture joint motion. Nevertheless, they are expensive, pose significant privacy concerns, and necessitate stringent illumination and placement settings.

The authors propose a millimeter-wave (mmWave)-based Movement Impairment Assisted Rehabilitati on System "MARS" to address these issues. MARS offers a cost-effective solution with comparable object localization and detection precision. MARS can reconstruct 19 human joints and their bones

using mmWave radar-generated point clouds. The system depicted in Figure.34 was developed by the authors using a Texas Instruments (TI) IWR1443 Boost mmWave radar [66]. For data acquisition, Texas Instruments' Matlab Runtime implementation was used. UART interface allows the device to communicate with the laptop. It begins retrieving data from the Matlab runtime with a 100ms frame duration.

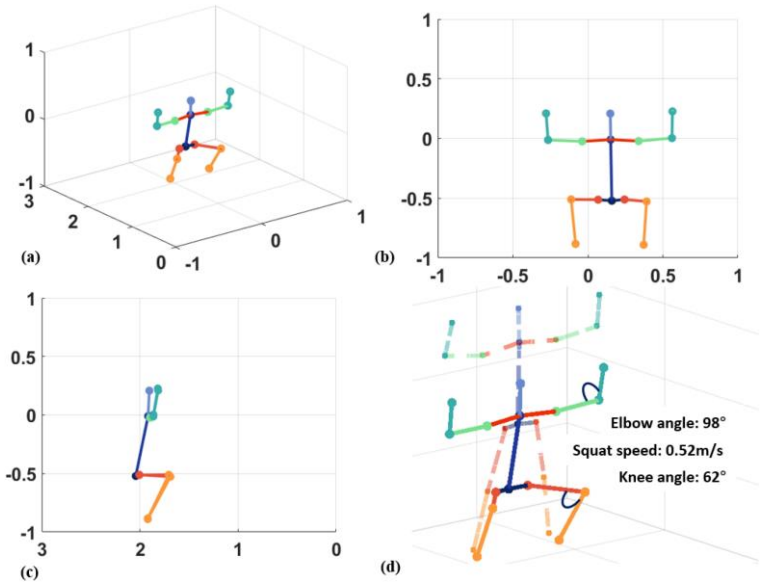

Figure (33). "Angle estimates displayed by the MARS system for the target squat movement"[10]

For various applications, the frame duration can be set to different values. Due to bandwidth limitations, they stipulated a minimum frame duration of 33.3ms, which corresponds to a sampling rate of 30 Hz. 100ms (or a sampling rate of 10 Hz) was chosen because it was adequate for measuring human motion ("most voluntary human motion occurs in the frequency range of 0.6 to 8 Hz" [84]). IWR1443 millimeter-wave radar [84] average power consumption at the power supply end is 2.1W. Kinect V2 sensor [85]: Using Microsoft Kinect V2, a ground truth reference is gathered. Both the Kinect and the radar were situated on a one-meter-high table, and the subjects performed the indicated tasks from a distance of two meters. Using an adapter, the Kinect V2 sensor is connected to a laptop's USB interface. Images are captured at a sampling rate of 30 Hz. Then, the images were processed with Matlab to identify the 3D coordinates of the 19 human joints enumerated in Table VI and demonstrate an average MAE of 5.87 cm for the MARS 3D joints position estimation.

This is the first open-source dataset of rehabilitation exercises using millimeter-wave point clouds, according to the authors. In addition, they intend to disseminate this dataset to the public via Github alongside the extant demo.

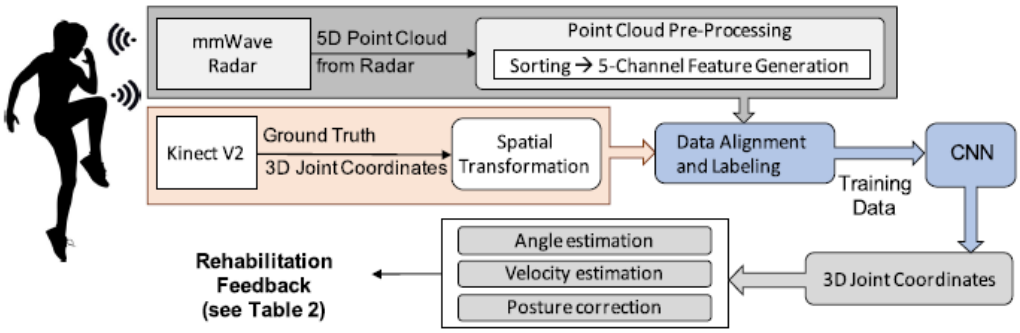

Figure (34). Overview of "MARS" framework [10]

7.4.1 Points cloud Pre-processing

The point cloud preprocessing in the "MARS" system includes the following steps:

(1) *Input data*: The main input of "MARS" is 5D time series point cloud. where is the x, y, z coordinates of the point reflecting the TX signal (x, y, z), the Doppler velocity D and the reflection intensity I.

Reformatting: The authors discuss the challenges associated with entering data, including:

Variation in the number of received points: If there are fewer body parts reflecting chirps, fewer than 64 points may be received. This leads to inconsistent input frame sizes.

Random order of points: Due to small variations in body posture and round-trip delay, reflected chirps arrive at the radar in a random order. This poses a challenge to the design of CNNs, which require fixed-shape inputs.

So, to address these challenges, the authors propose the following reformatting steps:

Padding: If less than 64 points are received, the rest of the frame is zero-filled to keep the input frame of uniform size (NP × 5) input frames.

Sorting: The points in each frame are sorted in ascending x, y, z coordinate order to solve the challenge of random point ordering.

These reformatting steps ensure that the input data is of consistent size and order, making it suitable for CNN processing.

(2) *Feature Generation (Ordering) for CNNs*: Due to slight variations in body pose and round-trip latency, the order of points in a frame is randomized. To solve this problem, the authors propose a preprocessing algorithm consisting of sorting and matrix transformation. Points are sorted in ascending order of x, y, and z coordinates.

(3) *Matrix transformation*: Transform the sorted input data with a size of 64 x 5 into a data structure of 8 x 8 x 5, in which five channels represent x, y, z

coordinates, Doppler velocity and reflection intensity. This transformation allows the data to be used in CNNs.

(4) *Dealing with "ghost images" out of range*: mmWave radar imaging can sometimes generate "ghost images" outside the range of interest in Figure.35(b)(d). The authors divide the point cloud into in-range points and out-of-range points, defining in-range points with specific boundaries for each dimension. Out-of-range points or ghosts are flagged and included during training and inference. (In order to prove that including the 'ghost map' works best, the author also conducted an Ablation Study).

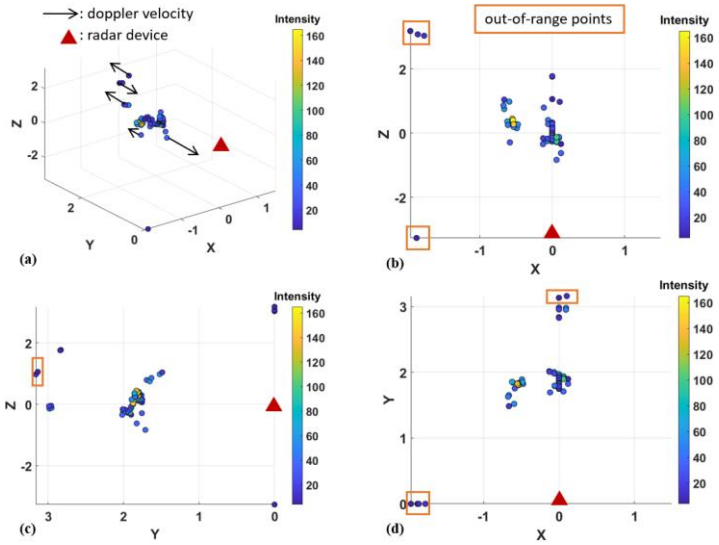

Figure (35). point cloud for one frame with ghost points. [10]

By including or excluding these out-of-range elements (approximately 2% of all frames), the authors evaluate the performance of the trained model. They trained two distinct CNN models, one with only point clouds within the range and the other with all point clouds, including points outside the range.

The results indicate that the model trained with all point clouds outperforms the model trained with only range point clouds. Incorporating out-of-range point clouds into the input increases the noise, making the CNN more robust. In addition, in real-world use cases, out-of-range points cannot be avoided. The authors concluded, based on these findings, that out-of-range elements should be included in the training data.

7.4.2 CNN (Convolutional Neural Network) model design

In MARS aims to convert the input 5-channel feature map into actual 3D joint positions, outputting the x, y, and z coordinates of 19 joints. The CNN model show in Figure.36 used in MARS is relatively simple and straightforward so i want to discuss are the author do an Ablation Study for CNN model design.

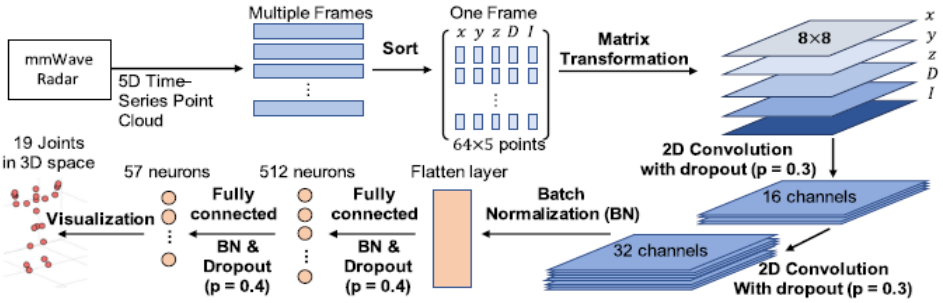

Figure (36). Point cloud pre-processing and CNN architecture.

Four different models are trained:

1. Baseline: no BN or max-pooling.
2. Baseline with BN: BN layers added after each convolution and fully connected layer.
3. Baseline with max-pooling: max-pooling layers added after the convolution layers.
4. Baseline with both: BN and max-pooling layers combined, with max-pooling layers added after each BN layer (except for the final BN layer after the fully connected layer).

Table VII. Average localization error of 19 joint-space locations for models trained with batch normalized and max-pooled CNN architectures [10]

	X (Horizontal) (cm)		Y (Depth) (cm)		Z (Vertical) (cm)		Average (cm)	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
Baseline	7.52	10.55	4.84	6.65	7.36	10.05	6.57	9.08
Baseline with BN	6.99	9.79	4.07	5.56	6.54	8.94	5.87	8.10
Baseline with max-pooling	7.63	10.38	4.65	6.64	7.15	9.57	6.48	8.86
Baseline with both	7.44	10.20	4.45	6.09	7.22	9.81	6.37	8.70

The results in this table are for 20% test data.

The results in Table VII show that the "Baseline with BN" model performs the best, while the "Baseline with both" model performs the worst, similar to the baseline model. BN is effective in avoiding the internal covariate shift, improving the model's performance. However, max-pooling is not a suitable option for this specific task, as it is a mapping regression problem that requires accurate joint coordinates. Max-pooling introduces information loss by taking the local maximum of features, making it difficult for the model to leverage every joint's coordinates effectively. Based on these findings, the MARS model incorporates only BN to achieve optimal performance.

In summary, the choice of using BN and not using max-pooling in the MARS model is based on the characteristics of the data and the specific requirements of the task. BN is used to stabilize the training process and improve performance, while max-pooling is avoided to preserve local information for accurate spatial

coordinate estimation.
At the same time, in order to prove that the 5D data works best, the author also conducted an Ablation Study on the input data features the results of Ablation Study show in Table VIII. The author set up 4 configuration situations.
Configuration-1: represents a CNN trained with only feature maps stacked by three channels x, y, z.
Configuration-2: represents a CNN trained with feature maps of x, y, z, and Doppler velocities; four channels stacked.
Configuration-3: represents a CNN trained with feature maps stacked with x, y, z, and reflection intensities.
Configuration-4: represents a CNN trained with feature maps of x, y, z, Doppler velocity, and reflection stacks strength.

Table VIII. Average localization error of 19 joint space positions obtained by models trained with different feature configurations [10]

	X (Horizontal) (cm)		Y (Depth) (cm)		Z (Vertical) (cm)		Average (cm)		
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	No. of parameters
Configuration-1	7.37	10.37	4.64	6.48	7.06	9.77	6.36	8.87	1,094,827
Configuration-2	7.33	10.20	4.37	6.02	7.00	9.52	6.23	8.58	1,094,971
Configuration-3	6.94	9.80	4.46	6.08	6.62	9.10	6.01	8.33	1,094,971
Configuration-4	6.99	9.79	4.07	5.56	6.54	8.94	5.87	8.10	1,095,115
mmPose[33]	6.80	10.21	4.79	6.67	6.94	9.86	6.18	8.91	2,281,739

The results in this table are for 20% test data.

They observe that the Configuration-1 model has the worst performance due to a lack of Doppler velocity and reflection intensity information, as shown in Table 5. Configuration-2 and Configuration-3 has slightly better performance since Doppler velocity or intensity information is introduced. Configuration-4 performs the best since the 5-channel feature maps contain all the information, including x, y, z with both Doppler and intensity. Note that because of weight sharing in CNN, adding channels in input only increases negligible parameters in the model, as shown in Table 5. They then apply Configuration-4 in MARS.

7.5 Summary for Using Multi-Dimensional Point Clouds

We now present state-of-the-art methods for human activity and gait recognition using multidimensional point cloud data, from the earliest mm-Pose [4] method that assigns RGB weights to 3D point clouds as input to a differential CNN model, to Meng and Z.'s article "Gait Recognition of Coexisting Multiple Persons Using Millimeter Wave Sensing" [5] uses 5-dimensional data X, Y, Z, plus radial velocity and signal strength, and "mmGaitNet" [5] multi-channel attribute network model, followed by It is the article written by Jiang X; Zhang, Y. et al. "Human gait

recognition based on millimeter-wave array radar using multi-channel 3D convolutional neural network" [7], these author uses the same 5D data input, 3D spatial coordinates, radial velocity and intensity as the 3-channel CNN model input, In the final "MARS: Intelligent Medical Assisted Rehabilitation System Based on Millimeter Waves" [10], the author also used 5-bit data input, but after data shaping and arrangement, the output is the same 3D data of human joints as mm-Pose .

7.5.1 Compare mm-Pose and MARS

In terms of accuracy, since the output of “mm-Pose” and “MARS” is not the classification result but the three-dimensional space data of the joints, the output of mm-Pose contains the three-dimensional data of 25 joints, while “MARS” contains the data of 19 joints, but the decrease in the number of joints does not determine the accuracy of the later reconstruction of the human body pose. Instead, what determines the accuracy of the later human body pose is whether the position of the joints in the three-dimensional space is accurate.

The result of mm-Pose is obtained when the feature map is 16x16 Got the results of “average mean errors of 3.2 cm in axis X, 2.7 cm in axis Z and 7.5 cm in axis Y, compared with state of art "MIT's RF-Pose3D"[21]”[10] on axis X and axis Z Better results were obtained by the authors, and the author setting the feature size of “MARS” into 8x8, then the author of “MARS” made a comparison after “reducing the size of the feature map to 8x8 in mm-Pose.” [10] The CNN model in MARS has about $1x10^6$ parameters, which only have half of the $2x10^6$ parameters in mm-Pose.

In addition, MARS' 3-axis positioning error has an MAE of 5.87 cm, which is less than mm-Pose's 6.18 cm. The results demonstrate that "MARS" feature generation decreases model complexity while increasing performance. Additionally, "mm-Pose" requires only two radars, whereas "MARS" requires only one, making it more practical and simpler to use. In addition, "MARS" is capable of handling sophisticated rehabilitation movements, whereas "mm-Pose" can analyze joint motion during walking.

7.5.2 Compare mmGaitNet & MC-3DCNN

For the other two models "mmGaitNet" in the experiment, if the radar is arranged diagonally, that is, scene 2, in the case of only two people walking, the highest recognition accuracy can be obtained. Of course, the accuracy will also decrease as the number of people in the scene increases,

In the "MC-3DCNN" experiment, "MC-3DCNN" also obtained an average accuracy of 93% after accumulating the accuracy of 4 cross-validation sets.

So, we can also make a summary of the methods proposed in these two articles, both of which take human gait recognition as the subject. "mmGaitNet" is a human gait recognition model specially designed for multi-person scenarios,

with a more complex model structure and input data. "MC-3DCNN" is a gait recognition model designed for single-person situations. Although the final accuracy of the two models is not very different, we found that both models have considerable limitations. For example, "mmGaitNet" requires two radars to be placed diagonally to obtain the maximum recognition accuracy. At the same time, if the number of people in an open space increase, the recognition accuracy will drop rapidly when the per capita gathering is the case. For "MC-3DCNN", although an average accuracy of 93% has been achieved, the system itself is designed for single-person gait recognition, and it is not suitable for deployment in open spaces. Yes, the usage scenarios of the system are greatly limited. Only suitable for use in spaces such as single patient rooms. Therefore, in terms of gait recognition, there is still a lot of room for improvement in research.

At this point we can finally discuss the following advantages and disadvantages of using multidimensional data.
By using multidimensional data, the system can achieve better performance in the joint estimation task because it utilizes more information about the scene. However, there are some trade-offs to consider:
Advantages:

1) More information leads to better joint estimation performance and accuracy.
2) Doppler velocity can help distinguish different body parts and their movements, which is especially useful for rehabilitation exercises.
3) Reflection intensity can provide additional insight into the scene, helping to improve the overall understanding of the environment.

Limitations:

1) This may require more processing power and time due to the increased computational complexity of the extra dimension.
2) More complex preprocessing and feature extraction techniques are required to effectively handle the increased data dimensionality.
3) The potential for overfitting, as more dimensions may increase the risk of the model capturing noise rather than meaningful patterns in the data.

Chapter VIII

Using Point Clouds and Range-Doppler

We have already introduced the method of directly using micro-Doppler features, the method of voxelizing 3D point cloud, and the method of using multi-dimensional point cloud data. All of these have one thing in common, which is to use single-form data for analysis. So, is it possible to use micro-Doppler features and 3D point cloud data for analysis at the same time? After searching, we found the method proposed by Huang, Y., Li, W, et al. We found their 2022 article "Activity Recognition Based on Millimeter-Wave Radar by Fusing Point Cloud and Range–Doppler Information"[15]. Next, I will introduce the method proposed in this article.

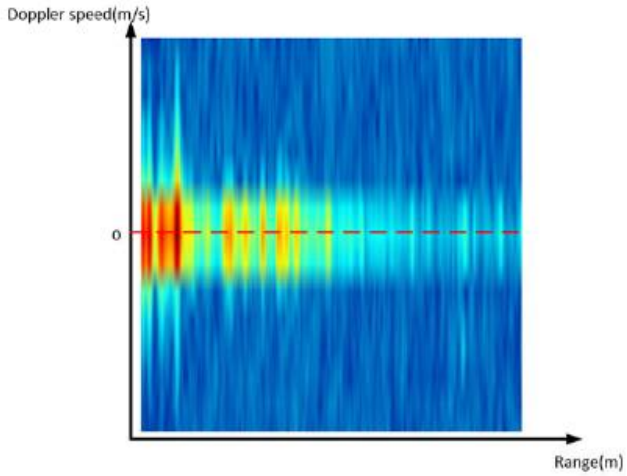

Figure (37). The visualization of range–Doppler data [15]

8.1 Activity Recognition Based on Millimeter-Wave Radar by Fusing Point Cloud and Range–Doppler Information [15]

The authors used the IWR1843 device, manufactured by Texas Instruments, for their data collection process. This device is a highly integrated, single-chip millimeter-wave sensor. It uses Frequency-Modulated Continuous Wave (FMCW) radar technology and operates within the 76 GHz to 81 GHz frequency band. The device incorporates three transmitting antennas, four receiving antennas, Analog-to-Digital Converters (ADC), and a Digital Signal Processing (DSP) subsystem.

Experiment setting: the device is linked to a computer and positioned at a height of 1.2 meters on a platform. The side of the device equipped with the antenna array is directed towards the individual performing the various activities. The positioning of the radar is crucial during data collection, and it's essential to ensure that the millimeter-wave radar remains fixed in place to avoid inaccuracies.

Six distinct activities are used in the data collection: boxing in place, jumping in place, squatting, walking in place, circling in place, and high knee lifting. These activities are executed by 17 different individuals, each performing the set of activities for 20 seconds.

The data collected are then labeled using a one-hot encoding technique. This is a process of converting categorical data into a form that could be provided to machine learning algorithms to improve predictions. Once labeled, the data is split into a training set and a validation set in a 4:1 ratio. This means that 80% of the total data is used for training the model, while the remaining 20% is used for validation purposes. The same dataset is utilized across all network models to ensure a fair and effective performance comparison.

8.1.1 Points cloud pre-processing

For the pre-processing of the entire point cloud data, the author adopts a very common approach:

(1) Denoising: The Cell Average Constant False Alarm Range (CA-CFAR) algorithm denoises the radar signal.

(2) Generate point cloud: Calculate the scale of the x, y and z axes. Create an array to represent 3D space. Populate the array with point cloud data.

(3) Reflect timing characteristics: In order to reflect the timing characteristics between frames, the author combines eight consecutive frames to add a time dimension, because the radar acquisition rate is 8 frames per second.

8.1.2 Ablation study for NN model and input data

In the research, the authors engage in detailed ablation studies to investigate the effect of various neural network architectures on classifying human activity. This is done by using both 3D point cloud data and range-Doppler signal. The ultimate aim of these studies is to demonstrate the advantages of training models using a fusion of 3D point cloud data and range-Doppler data.

3D point cloud data embodies the spatial and temporal attributes of movements. To extract spatial features, they deploy a three-dimensional Convolutional Neural Network (CNN). However, for temporal features, a straightforward CNN may not be entirely effective in capturing the sequential relationships between frames. To address this issue, they design two comparative network structures for their experiments. The first network strictly employs CNN, while the second network supplements the CNN with an LSTM

(Long Short-Term Memory) layer

3D CNN for 3D point cloud data

Initially, the authors develop a 3D Convolutional Neural Network (CNN) specifically for 3D point cloud data. This network is tailored to derive spatial features from the 3D point cloud data. Given the sparse nature of point clouds (typically fewer than ten-point clouds per frame), they amalgamate the point cloud data from eight successive frames into one, according to the collection rate is eight frames per second. They achieve this by summing up the respective three-dimensional arrays from the eight frames to form a denser, new three-dimensional array. This denser array is then utilized as the input for the convolution network training.

The network's architecture is depicted in Figure.38. For the convolution section, they employ a 'Conv+Conv+Max-pooling' setting. All the 3D convolution layers use a convolution kernel size of (3, 3, 3), with the kernel number of 32. Then set the padding to 'same', utilize the “Relu” activation function. Using the max-pooling layer and padding designated as 'valid'. Subsequently, the extracted features are flattened and inputted into the fully connected layer.

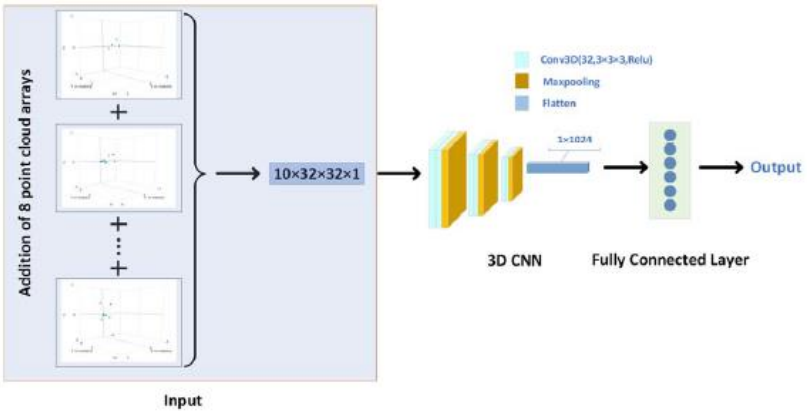

Figure (38). The structure of 3D CNN network [15]

For comparison, the authors develop a combined 3D CNN and Long Short-Term Memory (LSTM) model, also for 3D point cloud data. This design aims to accommodate the temporal relationships between frames by incorporating a bidirectional LSTM layer following the CNN. Consequently, they employ this hybrid 3D CNN + LSTM network for training and classification tasks. Given that the data collection rate is eight frames per second and each activity lasts about a second, they preprocess the 3D point cloud data into a four-dimensional array with dimensions 8 x 10 x 32 x 32. This includes merging 3D point cloud data gathered within a second.

As LSTM input must contain the 'timesteps' parameter, they wrap the convolutional layer, pooling layer, and flatten layer in a 'TimeDistributed' layer. The network structure is detailed in Figure 39. For the convolution section, they employ a 'Conv+Conv+Max-pooling' arrangement thrice, maintaining the same

parameter settings as in the initial network. The extracted features are then flattened and input into a bidirectional LSTM layer, equipped with hidden layer units. Lastly, after regularization, they employ a fully connected layer for classification, using the Softmax activation function.

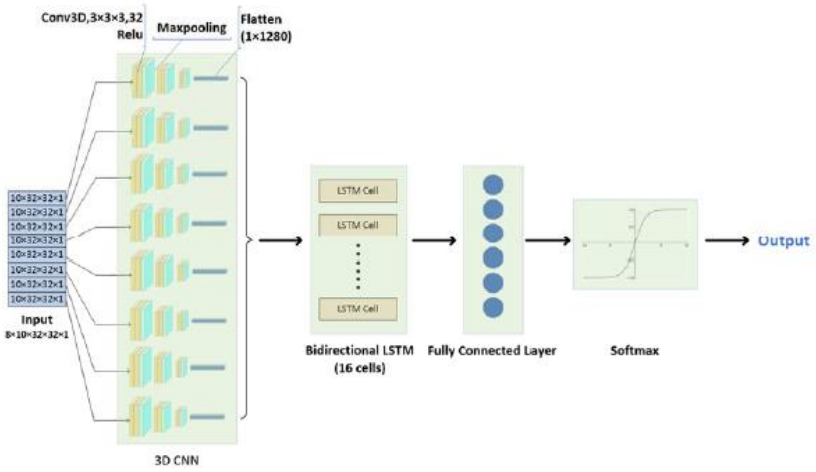

Figure (39). The structure of 3D CNN + LSTM network [15]

To validate their model, the authors design a 3D Convolutional Neural Network (CNN) specifically for range Doppler data. This network employs a 3D CNN to train on range Doppler data, treating the input as a three-dimensional single-channel stereo image. Since a single piece of range-Doppler data is essentially a temporal superposition of multiple images, the authors use a 3D CNN for training the range-Doppler data.

As depicted in Figure.40, this network shares a similar structure with the convolutional component of the point cloud network. The authors use the 'Conv3D-Conv3D-Max-pooling' structure three times, with parameter settings consistent with the network in preview part. However, this network does not utilize the 'Time-Distributed' layer wrapper to encapsulate the layers.

The features are then passed into the flattened layer and the fully connected layer consecutively for classification. This network processes eight frames of range-Doppler data as a single set of input data. In other words, data with the shape of 8 x 32 x 32 x 1 can be regarded as a three-dimensional single-channel stereo image. Therefore, this network can be perceived as a three-dimensional image classification network.

However, unlike traditional image classification networks, the input data possesses both spatial and temporal characteristics. The temporal characteristics can be interpreted as the series between frames, i.e., the features reflected by Doppler velocity. Consequently, this network can effectively learn the dynamic aspects of actions instead of merely extracting static two-dimensional features.

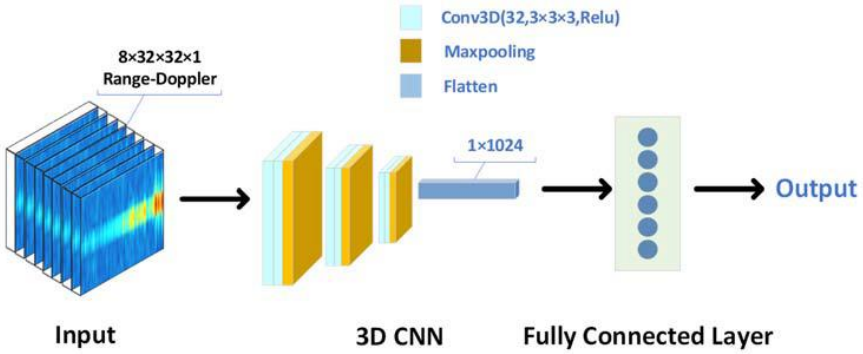

Figure (40). The structure of 3D CNN network for range Doppler data alone.[15]

The authors's base working principle is that the accuracy and detail of object classification can be enhanced when multiple types of information are used to describe that object, a concept known as multimodal fusion.

Various kinds of data, such as point cloud information, echo intensity information, and range-Doppler information, can be used to describe an activity. Each type of information provides a different perspective and unique characteristics about the activity. Therefore, extracting and effectively fusing these features can provide a more comprehensive view of the activity than using a single type of information.

The authors propose to merge the features from 3D point cloud information and range-Doppler information. They acknowledge that including more types of features could potentially offer a more complete picture of the activity. However, it would also increase the complexity of the fusion process. Therefore, they chose to focus only on the fusion of point cloud information and range-Doppler information.

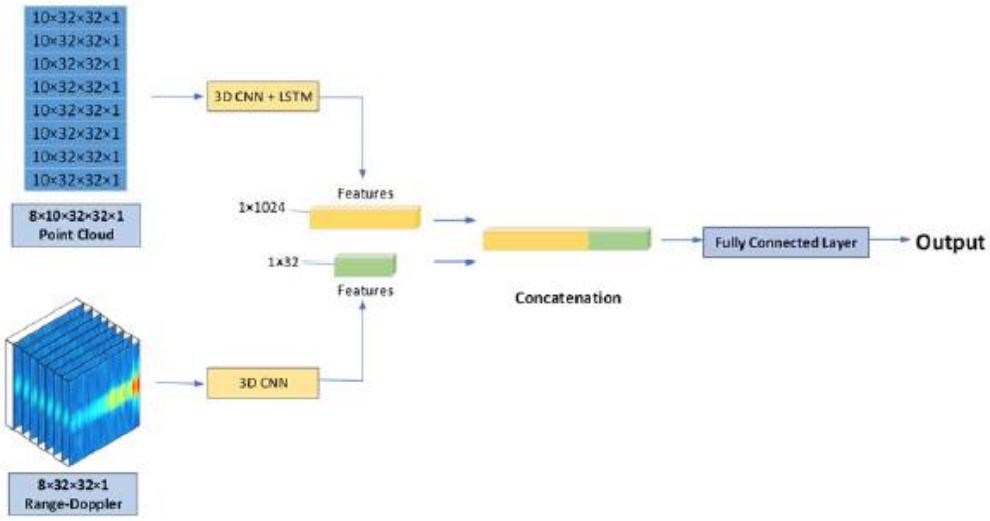

Figure (41). The structure of the fusion network.[15]

For this, the authors fuse the second and third models to design a fourth model,

a fusion network depicted in Figure.41. This network processes both 3D point cloud data and range-Doppler data. The inputs to the fully connected layers of the two networks represent the features inherent in these two types of data. The authors separately output the results from the preceding layer of the fully connected layer for each type of data, finding that the results are both two-dimensional, with the first dimension representing the volume of data.

These two sets of features are then fed into a concatenate layer, where they are merged along the second dimension to produce a feature set that contains both sets of characteristics. This combined feature set is then normalized and fed into a fully connected layer for classification.

8.1.3 Results and Discussion

In experiment, the authors tested two models using 3D point cloud data exclusively as input. The first model used only a three-dimensional Convolutional Neural Network (3D CNN), while the second model combined a 3D CNN with a Long Short-Term Memory (LSTM) layer, denoted as "3D CNN + LSTM". By analyzing the results presented in Table IX, it is clear that the "3D CNN + LSTM" model outperformed the simple "3D CNN" model in terms of accuracy. More specifically, the accuracy improvement of the "3D CNN + LSTM" model was significant, registering an increase of 6.59%.

This suggests that the addition of the LSTM layer to the 3D CNN model made a substantial contribution to the model's performance. The LSTM layer, a type of recurrent neural network, excels at processing time-series data or sequences, adding the ability to interpret the temporal dependencies in the sequence of 3D point cloud frames. Therefore, the improvement in accuracy likely results from the model's enhanced ability to comprehend the temporal dynamics in the data, thanks to the LSTM layer.

Table IX: Accuracy and Loss of The Two Networks Using points cloud as input.[15]

Network Structure	Accuracy	Loss
3D CNN	90.20	0.371
3D CNN + LSTM	96.59	0.129

At last, the authors compare three different models:

i. The "3D CNN + LSTM" model which solely uses 3D point cloud data,
ii. The "3D CNN" model which uses only range-Doppler data, and
iii. The fusion model which employs both types of data in a parallel manner, specifically, the "3D CNN + LSTM and 3D CNN" model.

By inspecting the accuracy curves visualized in Figure.42, the authors note that the fusion model, which utilizes both 3D point cloud data and range-Doppler data, converges fastest during training.

When comparing the performance metrics of accuracy and loss, the fusion

model outperforms the other two models. Specifically, the fusion model achieves the highest accuracy and the lowest loss compared to models trained on either type of data individually. This implies that the fusion of different types of data (3D point cloud and range-Doppler) within the model significantly enhances the model's predictive capability. The numerical results of the accuracy and loss for each of the three models are explicitly outlined in Table X, further substantiating the comparative analysis.

In essence, this section of the study highlights the advantage of using a multimodal approach in the data fusion model, which leverages the strengths of both the 3D point cloud and range-Doppler data. The incorporation of multiple types of data in parallel provides a more comprehensive feature representation, leading to superior model performance.

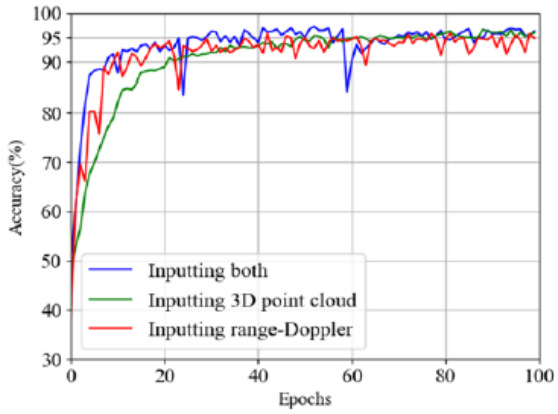

Figure (42). Accuracy and loss of the three different networks.[15]

Table X. Accuracy and loss comparison of three different networks with different input data [15]

Input Data Kinds	Network Structure	Accuracy	Loss
Only Range–Doppler	3D CNN	95.85	0.125
Only 3D point cloud	3D CNN + LSTM	96.59	0.129
Both 3D point cloud and Range–Doppler	3D CNN + LSTM and 3D CNN Parallelized	97.26	0.088

From the results, it can be seen that the fusion model of multiple data has great advantages in terms of convergence speed and accuracy, as well as model stability. Compared with models that only use a single data source, only some commonly used models are used for fusion. Very good results can be obtained afterwards as well. Therefore, we can think that the fusion model using multiple data will definitely have more advantages in terms of results than the model using a single data, and some scenarios that require high-accuracy system support will have great advantages.

But fusion models still have some limitations:

1. Complexity: The structure of the fusion model is more complex than that of a single network.
2. Overfitting: As the depth of the network increases, the risk of overfitting also increases.
3. Data dependency: The performance of fusion models is highly dependent on the quality and relevance of the input data.
4. Interpretation Difficulty: Interpretation and understanding can be more challenging due to the increased complexity of fused models.

8.1.4 Compare with "mmGaitNet"

Earlier we compared mmGaitNet and MC-3DCNN. "mmGaitNet" is a human gait recognition model specially designed for multi-person scenes, and the model structure and input data are more complex. "MC-3DCNN" is a gait recognition model designed for single-person situations. The method proposed by the author in this chapter is also specially designed for the single-person situation. Although the human gait is not recognized in the experiment, we can also make some comparisons with the single-person experiment. On the input data, "MC-3DCNN" adopts the form of mostly point cloud data, and the author of the article [15] adopts the fusion model "3D CNN+LSTM & 3D CNN Parallelized", the data input corresponding to this model The form is also two kinds of data, one is micro-Doppler feature data, and the other is 3D point cloud data. The two kinds of data are respectively input into the two models to generate output after correlation. In the model, it uses CNN as the baseline model like "MC-3DCNN". Here, the unique advantages of the CNN model in the classification of human activities are verified again, but if we look back at the article we discussed before [1] and [3], it is not difficult to find that the results obtained in the article [15] seem to be their fusion, [1] proves that CNN is very prominent in the analysis ability of Doppler signals, and the combination of CNN+LSTM The model's ability to analyze 3D point cloud data is exceptional.

From the results, "MC-3DCNN" achieved 93% accuracy in their respective experiments, and 97% accuracy in the article [15]. This is a very amazing result, but from the data and model in terms of the complexity of the article [15], two completely different data are used, and the differential fusion of the two models is used, which are far more complicated than "MC-3DCNN". This is not conducive to the expansion of future development in multi-person situations.

But in general, these two methods have verified the advantages of LSTM and CNN in human activity classification.

Chapter IX

Summary for Human Activity Recognition

Above we have introduced a variety of very representative methods in human activity recognition. These methods have promoted the development of human activity recognition using radar as a sensor. Next, I show more methods in Table XI below. , what is interesting is that these methods have used Texas Instruments radar equipment in the experiment, such as IWR1642 [100], IWR1443-Boost [66], IWR6843 [101], and IWR6843 [101] are all millimeter-wave radar applications in various fields It is widely used in scenarios, such as autonomous driving, etc., which also shows that the millimeter-wave radar equipment developed by Texas Instruments has better performance than other suppliers.
In the following Table XI, we can also obtain some interesting information. For example, in the classification problem of human activity recognition, the two models based on CNN model and LSTM are the most widely used, especially the spatial analysis ability of the CNN model makes it has better results and adaptability than other models, especially when facing multi-dimensional point cloud information.

From Table XI, we can also see that we have used CNN and LSTM models to achieve very high accuracy in single-person activity recognition, so in future research we can explore the use of different sensor fusion methods, such as infrared radar , and lidar, these radars can also have very high accuracy while ensuring privacy, and there are some models with very high popularity at this stage, such as Transformers, in which the ability to capture the relationship between different sensor inputs at different times may be will be reflected. These are the directions we need to study in the future.

Table XI. Comparison of Methods for Human Activity Recognition

Reference	Senser	Data Type	Deep Learning Model
[1]	IWR1642 [100]	Micro-Doppler signatures	CNN
[3]	IWR1443-BOOST [66]	3D voxel	Time-distributed CNN+ Bi-directional LSTM
[5]	2*IWR6843[101] 2*IWR1443[66]	5D point cloud	5 channels-Attribute Networks

[6]	4*TI AWR1243 [102]	2D point cloud	2D CNN
[7]	IWR1443Boost [66]	5D point cloud	5D Radar point cloud
[9]	4*AWR1243 FMCW [102]	3D point cloud	CNN + LSTM
[15]	IWR1843 [84]	3D point cloud + range-Doppler data	3D CNN+LSTM & 3D CNN Parallelized
[16]	IWR6843ISK-ODS [103]	3D Enhance voxel	Dual-View CNN (DVCNN) model

Chapter X

Human Posture Estimation

Pose estimation for human targets has been a popular area of study in recent years, similar to human activity recognition. Human activity recognition issues are fundamentally inseparable from pose estimation for human targets. In a sense, human activity recognition is a subset of human pose estimation. Nonetheless, among the numerous techniques for estimating the precise pose of the human body via mmWave radar, skeletal key point estimation has recently become a prominent research topic in the field of computer vision, identifying and detecting the human posture from images or videos.

Taking the information acquired by sensors and performing recognition can be used to provide information about the detected human posture, which is crucial for applications such as remote patient monitoring [85], because of the current shortage of medical personnel. Nevertheless, the aforementioned applications rely predominantly on optical sensing technologies, such as cameras and infrared (IR) sensors. Despite the fact that vision sensors provide a high-resolution description of the scene, their operation suffers under low-light conditions, inclement weather conditions, and when objects are obscured, which can result in recent, disastrous real-world consequences ranging from autonomous driving to car accidents. In addition, growing privacy concerns impede their practical application in patient monitoring systems.

Radar is operationally robust to scene illumination or weather conditions, albeit with a lower resolution scene representation than visual sensors. Furthermore, skeletal pose estimation using radar sensors has been relatively understudied. With high-bandwidth configurations, millimeter-wave (mmWave) radars can represent targets with greater resolution than conventional radar systems, but as a sparse point cloud in comparison to vision sensors.

However, the randomness between frames in radar point clouds makes explicit association difficult. Methods based on supervised machine learning (ML) can be used to identify and extract skeletal keypoint assignments from point clouds and to learn significant data characteristics.

The publications titled "mm-Pose" and "MARS" mentioned in Section 3.2.3 "Using Multi-Dimensional Millimeter-Wave Radar Point Clouds" are crucial to the estimation of bone key points.

Generally speaking, the method of skeletal key point estimation is as follows: firstly, after the 3D point cloud data of the human body is obtained through the millimeter-wave radar, the obtained point cloud data is preprocessed. The possible preprocessing method is to perform clustering algorithm on the point

cloud. Denoising and segmentation, then increase the density of the point cloud by sliding window method, etc., and then detect the selected joints by neural network.For example, 25 joints of the human body are selected in "mm-Pose", including: Spine (Base); Spine (Mid); Neck; Head; Left Shoulder; Left Elbow; Right Shoulder; Right Elbow; Left Hip; Left Knee; Left Ankle; Left Foot; Right Hip; Right Knee; Right Ankle; Right Foot; Spine (Shoulder).

In another article "MARS", the author selected 19 human joints, including: Spine Base; Spine Mid; Neck; Head; Shoulder Left; Elbow Left; Wrist Left; Shoulder Right; Elbow Right; Wrist Right; Hip Left; Knee Left; Ankle Right; Foot Right; SpineShoulder.

Also, in another article published by H. Cui and N. Dahnoun in January 2022:"Real-Time Short-Range Human Posture Estimation Using mmWave Radars and Neural Networks" [12], the author adopted 9 important joints, Left and right shoulders, left and right hips, left and right knees, left and right elbows, and head. But after the author collects the spatial positions of these 9 through the neural network, in order to improve the spatial reasoning between the joints, the author defines five main joints: head, left shoulder and right shoulder, and left hip and right hip. These joints are larger in size, produce stronger reflections of radar signals, and are more important for understanding the overall posture of a person.

Then the prediction of the minor joint depends on the adjacent major joint and the head. The authors use a dependency graph to represent these relationships between joints shown in Figure.43. Finally, the human pose is estimated by processing the input images and generating joint heatmaps.

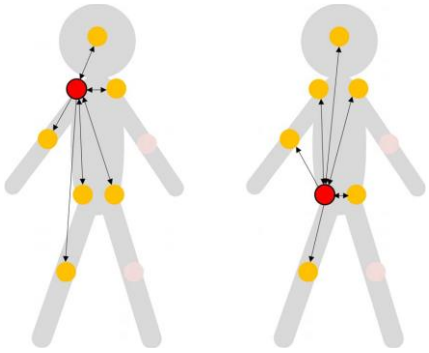

Fig (43). Dependency graph of the left shoulder and left hip.[12]

There are other bone key point estimation methods such as "m3Track" [13] proposed by Hao Kong, Xiangyu Xu, et al in June 2022. This method does not use point clouds, but uses range Doppler signals as Input data, but the author separates the original human body distance Doppler data into 6 Range-Angle-Profiles, including two head data information, two upper body data information and two lower body data information, plus 3 Range-Angle-Profiles Profiles for 'head cylinder'; 'torso cylinder'; 'leg cylinder'. Data partitioning is shown in Figure.44. Then the divided data is used as the input of a differential

CNN and LSTM fusion model. The output is the 17 joint positions selected by the author. Due to the advantages of range Doppler data in data size and calculation speed, the model can support 3D pose tracking and recognition of multiple people.

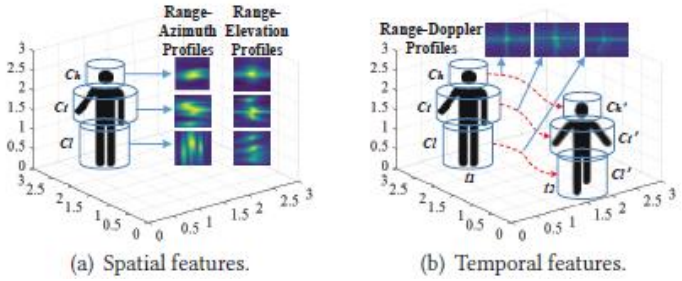

Figure (44). Illustration of posture feature representation for spatial and temporal features.[13]

When I introduced these methods, they were arranged in chronological order, so whether we can also find a trend that researchers are more inclined to use less key bone position information to reconstruct human posture. I think this makes sense. If less bone position information can make the model have faster processing power while achieving the same reconstruction accuracy, it may play a role in the future development of real-time pose recognition of multiple targets in open spaces. to a more important role.
But what we can find is that all the research so far requires researchers to collect radar point cloud data, which will essentially hinder the technological progress in this area for a certain period of time, and researchers also need to spend a lot of energy to choose Appropriate radar configuration parameters cause a certain amount of time wasted. So, the lack of available radar point cloud datasets makes further research and development more challenging.

10.1 “mmPose-NLP”[8]

Among the works proposing various skeletal keypoint estimation methods, methods for generating skeletal joint point cloud simulation data deserve special consideration. Alindam Sengupta et al. proposed "mmPose-NLP" [8], "a natural language processing (NLP)-based skeletal pose estimation method based on simulated millimeter-wave radar point cloud data." [8] The author's approach to obtain simulated data is to first obtain by adding noise to joint data acquired by Microsoft Kinect and then "using random sampling techniques to simulate the randomness and sparsity commonly observed in radar point clouds [8]". In the system, the author also uses voxelization technology, and "a unique 3D position generation index vocabulary [8]". Sequence-to-sequence (seq2seq) architectures [86][87][88] are used for “abstract summarization” of point clouds and

extraction of necessary skeleton keypoints.
According to the authors: "Besides their previous work on mm-Pose [4], this method is the only one that can estimate >15 skeletal keypoints in radio frequency (RF) based pose estimation" [8].

Through "mmPose-NLP" [8], future developers can develop many applications that benefit from human pose recognition technology, including autonomous vehicles, emergency rescue, real-time remote patient monitoring, (as shown in Figure.45, and security monitoring and national defense.

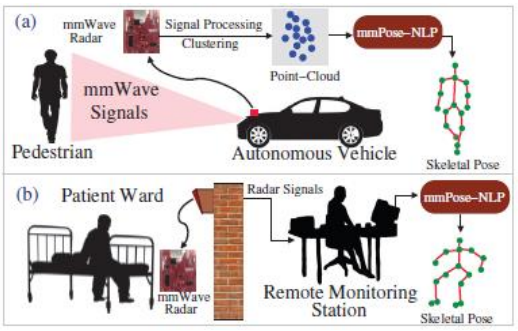

Figure (45). Possible application scenarios of mmPose-NLP: (a) autonomous driving; (b) remote clinical monitoring. [8]

10.1.1 Simulated points cloud generation processing

The whole processing involves several steps, including:
For generating simulated radar point-cloud data. Be more specific:

1. Obtain ground truth data of 25 human skeletal joints using Kinect's skeletal tracker on MATLAB API from every frame.
2. Corrupt the ground truth data with a random noise matrix.
3. Generate a noise distribution by adding synthetic points in a 3-D circular fashion around all possible joint links at a distance of 5cm.The example show in Figure.46.
4. Randomly sample a subset of 20 to 40 points from the corrupted ground truth data and noise distribution to obtain a simulated radar point-cloud.
5. Repeat steps 2-4 for all frames in the ground truth database to yield a sparsely simulated mmWave radar dataset.
6. Add Gaussian noise to the ground truth data to obtain noisy ground truth data.
7. Sample the noisy ground truth data to obtain the test dataset.

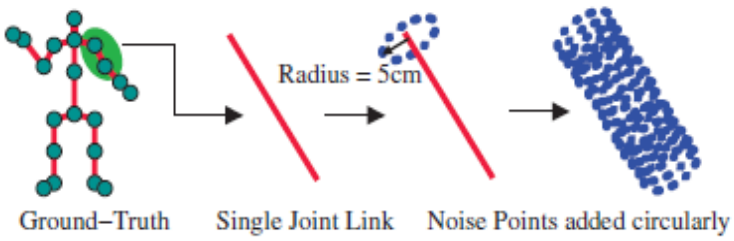

Fig (46). Points Cloud simulation flowchain [8]

We may ask the question why to use these methods to simulate human point clouds?

As i think the authors chose this process to generate simulated radar point-cloud from a human target because it emulates the randomness observed in radar point-cloud representation and provides a sparsely simulated mmWave radar dataset. Additionally, the use of Kinect's skeletal tracker and Mathworks developed skeletal tracking algorithm provides a reliable and accurate ground truth database for generating the simulated radar point-cloud.

Another possible approach could be to totally use a computer simulation software that models radar behavior and generates point-cloud data based on a virtual human model. This approach could be more flexible in terms of varying parameters such as radar frequency, target distance, and orientation. However, the accuracy of the simulation would depend on the quality of the human model and the accuracy of the radar behavior modeling.

In terms of which approach is better, it would depend on the specific requirements and constraints of the application. Using a physical radar system would provide the most realistic and accurate results but may be expensive and difficult to set up. Using a computer simulation may be more cost-effective and flexible but may not provide the same level of accuracy and realism.

10.1.2 “Seq2Seq Models” [86] [87] [88]

“Seq2Seq” is an acronym for sequence-to-sequence modeling, which is widely used in various natural language processing applications, such as keyphrase extraction, machine translation, and automated chat boxes [86] [87] [88].

The authors used the seq2seq method in the "mmPose-NLP" [8] architecture for abstract text summarization. Specifically, “The architecture uses as input a sequence of voxel indices generated by labeling 3D point cloud data obtained from mmWave radar. The input sequence is then processed by an encoder consisting of two layers of GRUs. processing, generating a compressed vector representation of the input. The GRU-based decoder then uses this compressed vector along with an attention mechanism to output a sequence of 25 voxel indices corresponding to the 25 skeletal keypoints being predicted [8]".

In summary, mmPose-NLP uses the seq2seq method to perform "abstract text summarization [8]" on millimeter-wave radar data, where the input data is a sequence of voxel indexes, and the output is a sequence of skeleton key points.
I think the advantage of the seq2seq method is that it can handle variable length inputs and outputs, making it a more flexible and general method than some alternatives. Furthermore, seq2seq models can be trained end-to-end, meaning that the entire model can be jointly trained to optimize the desired objective, rather than relying on handcrafted features or pipelines.

However, there are other approaches to NLP tasks besides seq2seq, and the best approach may depend on the specific task and dataset. Some alternatives include:

Transformers [93]: These models, such as BERT [89] and GPT [90][91][92], use self-attention mechanisms to model the relationship between different parts of the input sequence. They achieve state-of-the-art performance on many NLP tasks.

Recurrent Neural Networks (RNN): These models use tree structures to represent input sentences and can capture complex dependencies between words.

The best approach may depend on the specific task and dataset. For example, a seq2seq model may be better suited for tasks where the input and output sequences are of different lengths, while a Transformer model may be better suited for tasks where the input and output sequences are of similar length. Ultimately, the choice of method will depend on the specific requirements of the task and the available data.

10.1.3 "mmPose-NLP Architecture"[8]

The mmPose-NLP architecture is used for abstract text summarization of simulated mmWave radar data in order to extract the desired 25 skeletal key-points from Figure.47's overall structure. Similar to tokenization in NLP preprocessing, the procedure entails the creation of a vocabulary dictionary to map each 3-D point (x, y, z) to a distinct integer. This is accomplished by creating a voxel space with dimensions of 1.7 m 2.2 m 1.2 m, with voxels measuring 5 x 5 x 5 in centimeters. Every 3-D coordinate in the voxel space is associated with the voxel that contains it and tokenized with the voxel's integer index.

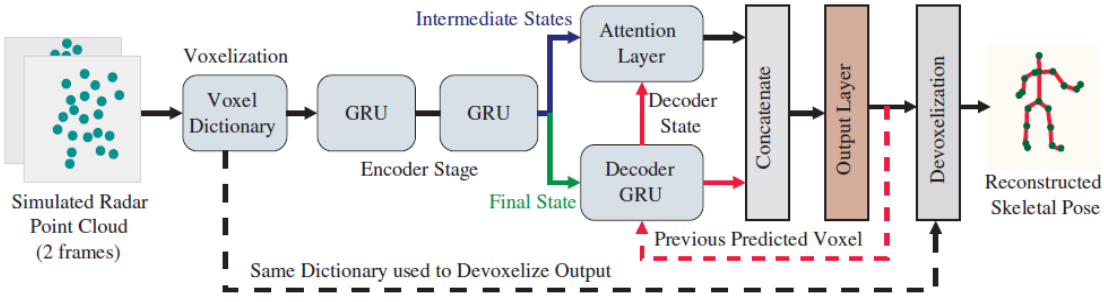

Fig (47). "mmPose-NLP architecture. [8]

In this system, two consecutive frames of voxelized data are fused as one input to mmPose-NLP. This sequence is then projected into the representation using an embedding layer. The authors then use a two-layer GRU encoder to form a model that expresses the input sequence by compressing the vector. When the system receives a "START" command, "the GRU-based decoder will use this encoded representation to update its current internal state [8]". The attention

layer then takes these two types of information (decoder state and weighted encoder hidden state) and generates attention vectors. The attention vector essentially determines which parts of the input data the decoder should focus on. The attention vector and decoder state are then combined and fed to a fully connected output layer. A fully connected layer is the part of a neural network where each neuron is connected to every neuron in the next layer. This output layer predicts specific voxel indices. Voxels are equivalent to 3D pixels. In this case, it can represent a point in 3D space that corresponds to a part of a human skeleton. This predicted voxel index is then fed back into the decoder as the next input, and the process is repeated until all 25 skeleton keypoints or the desired number of words are predicted. Each keypoint represents a joint or an important point on the human body whose position the model is trying to estimate.
After obtaining the 25 voxel indices, the authors can de-voxelize them using the voxel dictionary used in the tokenization process and finally represent them back in 3D coordinates as a point cloud.

10.1.4 Result & Discuss

The authors generated simulated radar-like point cloud data from ground-truth skeletal data, adding 3D gaussian noise to simulate measurement errors. Nine datasets were created with varying levels of noise (σ) and were used to test the model's ability to reconstruct the actual skeletal pose. The model showed lower localization errors compared to existing approaches like mmPose and RF-Pose. The study also showed the mmPose-NLP's achievable performance for a given mmWave radar resolution. The work provides a link to the code and data for further exploration. The result show in the Table XII.

TABLE XII

LOCALIZATION ACCURACY COMPARISON

	Localization Error (cm)		
Method	Horizontal	Vertical	Depth
mmPose-NLP (σ = 0.1cm)	1.19	0.96	1.38
mmPose-NLP (σ = 0.25cm)	1.22	0.99	1.40
mmPose-NLP (σ = 0.5cm)	1.31	1.11	1.53
mmPose-NLP (σ = 0.75cm)	1.46	1.23	1.64
mmPose-NLP (σ = 1cm)	1.57	1.31	1.73
mmPose-NLP (σ = 2cm)	2.11	1.69	2.22
mmPose-NLP (σ = 3cm)	2.59	2.07	2.77
mmPose-NLP (σ = 4cm)	2.98	2.40	3.17
mmPose-NLP (σ = 5cm)	3.32	2.64	3.61
mm-Pose [10]	7.50	2.70	3.20
RF-Pose (15 Points) [24]	4.90	4.00	4.20

Both “mm-Pose” and “mmPose-NLP” are methods for estimating human skeletal posture using mmWave radar data and deep learning techniques. However, “mmPose-NLP” has some advantages and limits compared to mm-Pose.
Advantages of “mmPose-NLP” are:

First “mmPose-NLP” uses natural language processing techniques to incorporate semantic information in the input data, which can improve the accuracy of the skeletal posture estimation. Then “mmPose-NLP” introduces a novel method for generating simulated mmWave radar-like point cloud data, which can help in training and testing the model in a controlled environment. And “mmPose-NLP” achieves lower localization errors compared to mm-Pose and RF-Pose in some experiments.
Limits of “mmPose-NLP”:
“mmPose-NLP” requires a larger amount of training data compared to mm-Pose, as it needs both skeletal joint information and natural language descriptions.
“mmPose-NLP” is limited to estimating skeletal posture of humans and cannot be used for other objects or animals.
“mmPose-NLP” may have limitations in noisy environments or when the target is occluded or partially visible.

Overall, “mmPose-NLP” introduces some novel techniques and achieves promising results in some experiments, but it also has some limitations that need to be addressed.

Earlier we introduced "mmPose-NLP"[8] a “natural language processing (NLP)”[95] based approach for skeletal pose estimation from simulated mmWave radar point cloud data. Under such circumstances, the point cloud data of the human body can be obtained directly through simulation, which can greatly reduce the time for designing and arranging the test site and collecting experimental data. To the best of our knowledge, it is also the only method that can simulate point clouds of more than 15 skeletal joints. We have introduced many methods for human pose recognition through skeletal joint localization. There are methods of directly locating multiple skeletal joint positions through multi-dimensional point clouds [4][10], and there is also a method of locating important skeletal joint positions through 2D-garyscale images to reconstruct postures [12]. There is also a method of converting distance Doppler data into 2D and 3D tensor data training depth model to obtain the method of bone joint position [13]. But is there a fast and scalable human pose estimation framework that can be adapted to various machine learning models? Sizhe An and Umit Y. Ogras. al. proposed a method "FUSE". I will introduce it next.

10.2 FUSE [11]

FUSE propose by the author is a human pose estimation technique that uses mmWave point cloud data. It consists of two main components:
1. Fusing sparse frames to build data representations for multi-frame fusion.
2. A meta-learning framework capable of adapting to unfamiliar data in a fraction

of the time. But the important is that: Meta-learing framewok here cloud compatible with a variety of neural network models as long as the purpose of point cloud analysis can be achieved. Like (CNN, GCN, Pointnet ...). The framework of FUSE show in Figure.48.

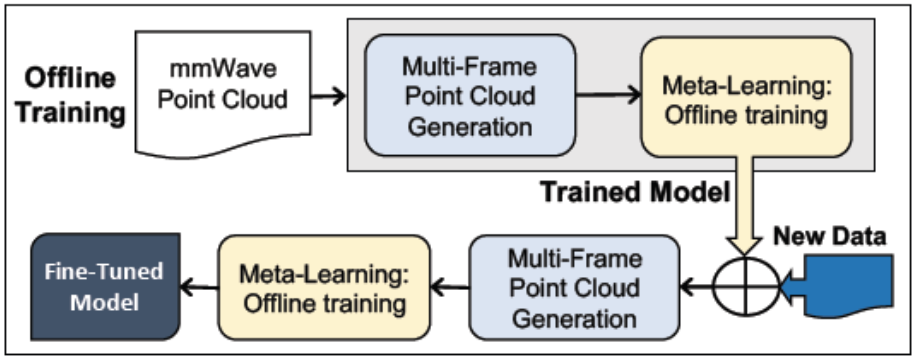

Figure (48): "FUSE" framework overview [11]

(1) Baseline model: In the FUSE work, the authors propose a new model for handling human activity recognition tasks. They start by defining a baseline model that serves as a reference point for evaluating the improvements offered by the FUSE framework. The baseline model is a Convolutional Neural Network (CNN), which is a type of deep learning model widely used for analyzing visual imagery.

The baseline CNN consists of two layers of convolutional neural networks followed by two fully connected (FC) layers. The convolutional layers are designed to automatically and adaptively learn spatial hierarchies of features from the input data. The Rectified Linear Unit (ReLU) activation function is applied in these layers to introduce non-linearity into the model, which allows the network to learn complex patterns.

The fully connected layers serve to perform high-level reasoning based on the features extracted by the convolutional layers. The first FC layer consists of 512 neurons, while the second one has 57 neurons. These 57 output neurons represent 19 human joint coordinates in the x, y, and z axes, as each joint coordinate is represented by three values (x, y, and z). Parameters in a neural network model are the internal variables that the model adjusts through learning, enabling it to better fit the input data.

Subsequently, the authors apply the FUSE framework to this baseline model. The FUSE-enhanced model maintains the same dimensions and model size as the baseline model to ensure a fair comparison of their performances. The implementation of the meta-learning approach used in FUSE is based on the MAML-PyTorch [94] framework.

In summary, the baseline model is a CNN with two convolutional layers and two FC layers. The FUSE framework is applied to this baseline model to enable quick adaptation to new users or movements with only a few training samples.

(2) About the points cloud pre-processing: (Multi-Frame Fusion of Point Cloud Data)

The authors of the FUSE work confront the challenge of sparse millimeter-wave (mmWave) point cloud data. Unlike traditional video frames, which have an abundance of data points (each pixel in the frame being a data point), mmWave point cloud data contains significantly fewer data points. This sparsity can hinder the effectiveness of machine learning algorithms, as there may not be enough information to accurately train models and extract valuable insights.

To address this challenge, the authors propose fusing or combining multiple sparse point cloud frames to generate a denser or richer representation of data. This concept is inspired by the notion of residual frames used in video processing. Residual frames, in the context of video processing, emphasize the changes between frames that are due to motion, therefore reducing redundancy. However, the authors' goal is not to reduce redundancy, but rather to increase the data density in the sparse mmWave radar data.

The authors use a time interval, referred to as a sampling period (Ts), and fuse M consecutive frames by concatenating them. The number of fused frames is controlled by the parameter M. For instance, when M equals 1, three frames are fused: the current frame, the previous frame, and the next frame. The authors show in Figure.49(d) that their proposed multi-frame fusion approach greatly enhances the interpretability compared to using a single mmWave point cloud frame.

In order to show the stronger interpretability of the information acquired by the multi-frame point cloud representation, when the author compares the multi-frame representation with the single-frame representation (as shown in Figure.49(b), it is obvious that the multi-frame representation method can be more accurate Accurately captures the shape of the upper body. The authors observe that there are more data points around the subject and arm regions in the multi-frame representation.

The processing flow should be like this:

First every frame of the point cloud (the set of data points that the radar collects) corresponds to a certain time interval. This time interval is controlled by the sampling period, Ts, which is set at 100 milliseconds.

So, the k^{th} frame of data represents all the points collected in the time between k*Ts and (k+1) *Ts. The k^{th} frame can be written as:

$$f[k] = [P_1[k], P_2[k], ..., P_N[k]] \forall k \subseteq \mathbb{Z}^+ \quad (24)$$

Then each frame can be represented as a set of points,

They then combine M consecutive frames by concatenating them, creating

a richer set of data. This combined frame is represented as

Where M is a parameter that determines how many frames to fuse together. For instance, if M=1, three frames are fused together: the current frame, the previous frame, and the next frame.

So, their proposal could significantly improve previous findings using millimeter wave point cloud data.

The authors of FUSE propose an innovative method to handle the sparsity inherent in millimeter-wave (mmWave) point cloud data: fusing or combining multiple frames. This technique substantially enhances the amount and interpretability of the data, making it more conducive for machine learning algorithms to extract relevant features. More specifically, the multi-frame representation provides a more accurate depiction of the subject's shape, which is a crucial aspect when estimating human poses. This enhancement can lead to improved performance in pose estimation models.

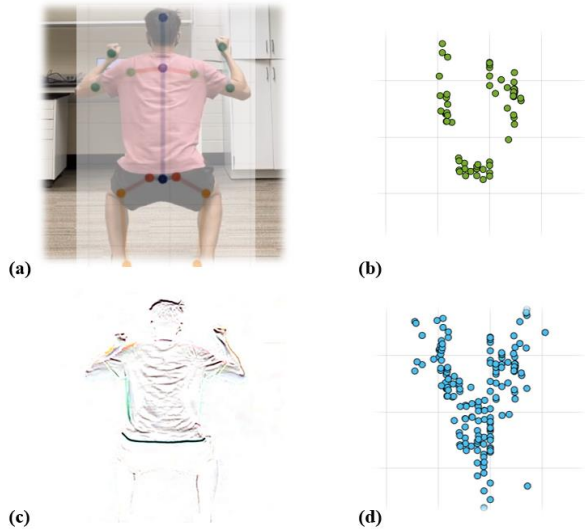

Figure (47): (a) shows the RGB image frame; (b) shows the collected single-frame point cloud; (c) shows the RGB residual frame; (d) shows the visual image of the multi-frame point cloud proposed by the authors [11]

However, there are a few potential challenges associated with this approach:
Selection of the Parameter M: This parameter determines the number of frames to fuse, and its value could significantly impact the model's performance. An improperly chosen value could lead to poorer performance or additional computational demands.
Computational Requirements: While it isn't explicitly stated in the text, it's worth noting that fusing multiple frames could lead to an increase in computational requirements, as the model now has to process more data points per input.

To determine the most effective configuration, the authors conducted experiments under three different settings:

i. Using a single frame (serving as the baseline comparison).
ii. Using three frames together.
iii. Using five frames together.

The experiments' results revealed that fusing three frames resulted in a consistent decrease in the Mean Absolute Error (MAE) along the x, y, and z axes. Specifically, by fusing three frames, the average MAE reduced from 5.5 cm to 3.6 cm, demonstrating a considerable improvement of 34%. However, fusing more than three frames didn't lead to further enhancement, as it started introducing redundancy, meaning the extra data wasn't providing additional useful information.

The multi-frame fusion pre-processing technique improves the performance of human pose estimation tasks by providing a richer data representation. This enhancement can boost the performance of existing mmWave radar techniques without affecting the machine learning models they employ. In their experiments, the authors chose to fuse three frames, as it led to significant improvements with negligible overhead.

10.2.1 Meta-Learing processing

In the FUSE framework, meta-learning is used to enhance human pose estimation by enabling the model to rapidly adapt to new users and movements using a small number of training examples. The FUSE meta-learning procedure consists of two distinct phases: offline meta-training and online fine-tuning. The Algorithm: Meta-training for mmWave point cloud show in Figure.50.
First, in order to avoid confusion and respect the authors' work in this section, I will use the authors' definitions of the parameters and terms used in the meta-learning models they designed, and present their definitions below:

"Definition 1 (Training data, D$train$). The training data is the set of all fused frames F [k], $k \geqslant 1$ constructed using the point cloud frames as defined by Equation 3, i.e.,

$$\boldsymbol{D}_{train} = \cup_k F[k]$$

Instead of directly using individual samples in D$train$, meta-learning generates tasks and uses them for learning as described next.

Definition 2 (Task, $\mathcal{T}$). We define task $\mathcal{T}$a set of fused frames uniformly sampled from the training data, i.e., $\mathcal{T} \sim \mathcal{D}train.$ "[11].

Next, we present the proposed offline meta-training and online fine-tuning

techniques.

Offline Meta-Training: In this phase, the model is initially trained with tasks generated from the training dataset ($\mathcal{D}train$). The model's parameters (Θ) are first randomly initialized. Then, through iterative meta-training, these parameters are updated. During each iteration, a batch of tasks is sampled from the training data, and the model parameters are then updated via gradient descent, a common optimization technique in machine learning. The crucial part of this phase is how the initial parameters are updated. Instead of directly using the results from the tasks, the model takes the intermediate parameters derived from the "support" tasks (part of the batch that is used to update the model) but evaluates the loss using the "query" tasks (a separate part of the batch that is used to evaluate how well the model is learning). This process helps the model identify parameters that are most sensitive to new data samples, thus improving its ability to adapt to new tasks.

Online Fine-Tuning Phase: Following the construction of the initial meta-learned model, the authors' goal is to adapt the model to handle a new user or movement using a small set of test data ($\mathcal{D}t$est). In this phase, the model is fine-tuned using part of $\mathcal{D}t$est, and then its performance is evaluated using the remaining part of $\mathcal{D}t$est. This fine-tuning phase does not require any additional steps and enables straightforward online usage, allowing the model to quickly adapt to new users or movements.

Input: $\mathcal{D}_{train}$, g_θ (untrained model), β (meta-learning rate)
Output: ML model that computes human joint coordinates using mmWave point cloud.

Initialize the parameters θ of the ML model g_θ
for *each meta-training iteration* **do**
  Sample a batch of tasks: $\mathcal{T}_i \sim \mathcal{D}_{train}$
  for *all* $\mathcal{T}_i$ *do* **do**
    Sample support tasks from $\mathcal{T}_i : \mathcal{T}_i^{sup} \subset \mathcal{T}_i$
    Compute the gradient $\nabla_\theta L_{T_i^{sup}}(g_\theta)$
    Update parameters $\theta_i' = \theta - \alpha \nabla_\theta L_{T_i^{sup}}(g_\theta)$
    Sample query tasks $\mathcal{T}_i^{qry} \subset \mathcal{T}_i$
    Evaluate $L_{T_i^{qry}}(g_{\theta_i'})$ using parameters θ_i'
  end
  Update the initial parameters
  $\theta = \theta - \beta \nabla_\theta \sum_{\mathcal{T}_i^{qry}} L_{\mathcal{T}_i^{qry}}(g_{\theta_i'})$
end

Figure (50). Algorithm: Meta-training for mmWave point cloud [11]

The use of meta-learning in the FUSE framework offers several advantages, including rapid adaptation to new users and movements, reduced data requirements compared to traditional supervised techniques, and easy online usage. However, limitations include potential computational expense during the training phase and the dependence on the diversity and quality of the training tasks. Overall, the incorporation of meta-learning in FUSE enables the model to swiftly adapt to new situations using fewer training samples and iterations, enhancing the performance of human pose estimation.

10.2.2 Convergence Time and Accuracy Evaluation

The authors in the "Convergence Time and Accuracy Evaluation" section, assess their FUSE framework's capability to swiftly and efficiently adapt to new situations. They design an experiment to compare the FUSE performance with the baseline model.

Human Pose Estimation Data: For the evaluation, they utilize an open-source millimeter-wave point cloud dataset (MARS). This dataset comprises 40,083 labeled frames collected using TI IWR1443 Boost [66] millimeter-wave radar. These frames represent ten distinct rehabilitation movements executed by four individuals. Using a Kinect V2 sensor, the reference coordinates of 19 joints are determined and appended as identifiers to the millimeter-wave data sampled at 10 Hz. Then, the data for each movement is divided into 60% training sets, 20% validation sets, and 20% test sets.

To scrutinize FUSE's capability to adapt to new scenarios, they manipulate the dataset to simulate a worst-case scenario. The training and validation sets exclude all data from a particular movement ("right limb extension") and the user No.4. The test data ($\mathcal{D}$test), therefore, seen only during fine-tuning, has only 749 frames, which helps support their claim of online adaptation with a few samples. In contrast, the training data ($\mathcal{D}train$) comprises 29,225 frames from the remaining movements and users.

Fine-tuning, a common method used in transfer learning, is then employed to assess the model's adaptability to new data samples. They conduct tests for both instances: All layers are fine-tuned, but only the final FC layer is activated.

"Fine-tune all layers" [11]: The FUSE model converges rapidly, attaining a Mean Absolute Error (MAE) of approximately 6.0 cm with the new data after only 5 epochs. The baseline model, on the other hand, requires at least 20 epochs to attain comparable performance, at the expense of forgetting the original data. Display in Figure.51.

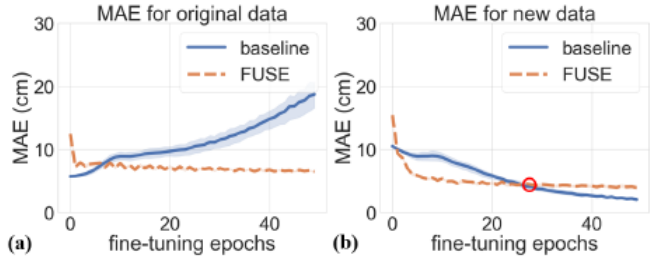

Figure (51): "MAE comparison between baseline and FUSE model for fine-tuning all layers." [11]

"Fine-tune the last layer" [11]: The FUSE model achieves a MAE of 8.3 cm after only 5 epochs, 1.3 cm lower than the baseline. Show in Figure.52. The baseline model achieves a similar performance as the FUSE model after 16 epochs, but again, it forgets the original data in the process.

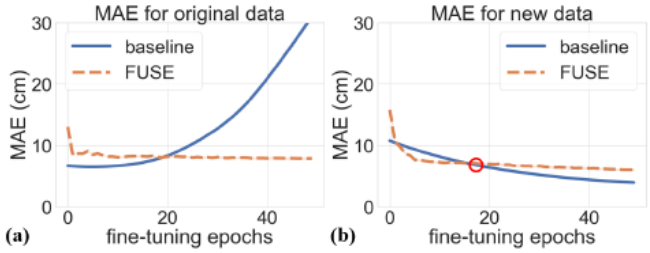

Figure (52): "MAE comparison between baseline and FUSE model for fine-tuning the last layer." [11]

The results demonstrate that the FUSE framework improves human pose estimation performance and quickly adapts to unseen data. It converges about 4 times faster than the baseline approach, without forgetting the original data.

Chapter XI

Summary for Human posture Recognition

In the following Table XIII, I compared several methods of human body pose recognition. We can see that in the neighborhood of human body pose recognition, researchers invariably used the method of human body Google key point positioning to reconstruct human body poses. Due to millimeter-wave radar It is difficult for us to directly analyze the posture of the human body from the obtained point cloud, especially in the absence of an open-source database of human millimeter-wave radar point cloud at this stage. In addition, by observing the table below, we can also find that in the case of using directly collected radar data, the higher the dimensionality of the data used by the researchers, the higher the spatial accuracy of the final key bones, which shows that the direct use of experimental data in the case of data, the more data information and the accuracy of the final result are positively correlated. In addition, the simulated point cloud method adopted in [8] achieved the highest accuracy. The purpose of this method is to simulate the millimeter-wave radar human body point cloud by computer simulation under the condition of lack of radar equipment. In this method the point cloud information simulated in is usually better than the data collected by radar in terms of noise and point cloud sparsity.

In addition, since our table is arranged in chronological order, we can observe that over time, researchers tend to use fewer human skeleton key points for human pose reconstruction, which I think makes sense, which will be more suitable for human gesture recognition in the future when the amount of data is increasing in the case of multiple people.

Table XIII. Comparison of Methods for Human Skeleton Reconstruction

Reference	Senser	Data Type	Number of joints	Resolution
[4]	2*AWR1642 Boost [100]	3D points cloud	25	6.18cm (MAE)
[8]	N.A.	Simulated mmWave-radar points-cloud data	25	5cm (MAE)

[10]	IWR1443 Boost [66]	4D points cloud (Heat map)	19	5.87cm (MAE)
[11]	IWR1443 Boost [66]	3D point cloud	19	6cm (MAE)
[12]	2*IWR1443 Boost [66]	2D grayscale image	9	12cm (MLE)

Chapter XII

Conclusion

The purpose of this work is to introduce existing representative analysis techniques for point cloud data collected using mmWave radar as a sensor in the fields of human localization, human tracking, human activity recognition, and human pose recognition.

But it is worth reminding again that the purpose of this work is not to screen the best existing methods, but to sort out the current technical status and existing technical ideas in the existing neighborhood.

Second, when comparing similar methods, this work draws a fundamental distinction based on the subject matter of the methods.

We can draw some very interesting results after analyzing and comparing.

First of all, we found that in order to obtain high-precision results in human activity recognition, first of all, good experimental equipment and perfect experimental environment settings are the prerequisites for all experiments to obtain good results. Secondly, in terms of methods, some preprocessing methods that can improve the portability of data information will have a great impact on the results, such as the "point cloud voxelization" we introduced [2], [3], [16]. Of course, There are many methods and similar measures that I will not point out here. But we can see the superiority of "voxelization" measures, such as data interpretability and data readability. Similarly, in order to compensate for the loss of data features during the voxelization process, researchers also need to do some measures such as multi-frame fusion [3] to make up for it.

In terms of models, we can also see that the authors have chosen variant models based on CNN and LSTM, such as "Bi-LSTM" used in [2] and "Time distributed CNN+ Bi -directional LSTM". In [16], the author used a differential CNN model. It is not difficult to see that the voxelized data is the best choice to use CNN and LSTM models that are very sensitive to spatial and temporal features.

Of course, in addition to "voxelization", there are many methods of using "multi-dimensional point cloud" data [4], [5], [7], [10]. The most intuitive advantage of multi-dimensional point cloud data is that the data carried the information is rich. Generally speaking, in addition to the three-dimensional coordinate information, there is also information such as speed and signal strength. This information can greatly help researchers restore the subjects' body, activity and other information in the experiment, and finally achieve the purpose of personnel detection, activity recognition and gesture recognition. But the obvious disadvantage is that multi-dimensional data usually has a huge

amount of data, and general models cannot process so many different types of information at the same time. Therefore, in the preprocessing stage, experimenters need to use signal denoising, point cloud operation, point cloud Clustering and a series of means to filter and clean data. Similarly, it takes a lot of energy to design the model. The LSTM with a relatively single processing capacity is basically not selected by the experimenters. All the experimenters have chosen the baseline model of CNN. At the same time, a large part of researchers has chosen Using the differential model, for example, the authors in [4], [5], and [7] used the difference on the XY and XZ planes, the differential channel distinguished by feature attributes, and the 3-D spatial coordinates, radial Velocity, and intensity are 3-channel designs distinguished. These methods finally achieved very good test results.

Finally, there is a method of fusing micro-Doppler features and point cloud data, and the results obtained by combining CNN and LSTM models are also very impressive.

In addition, in the field of human activity detection and gesture recognition, the lack of open-source data sets is always a very difficult problem. Unlike lidar, cameras and other equipment, millimeter wave radar experimenters need to spend a lot of time and effort to collect experimental data. Therefore, the method proposed in the similar article [8] is particularly precious, a method that can obtain human body millimeter-wave radar data through computer simulation. There is also the "3D Body Reconstruction Dataset" made by the author of the article [14], which will also greatly facilitate researchers to conduct experiments in the future.

I think the recognition of human activities and gestures in the future needs to solve the following problems:

1. More reliable open-source datasets.
2. For more than 10 people, research on human activities and postures
3. Research on compatibility with unfamiliar environments.

In general, this work is to facilitate subsequent researchers who want to continue to study this research field to have a quick and comprehensive understanding of the current state of the art.

I think the future of human activity and gesture recognition needs to solve the following problems:

1. More reliable open source datasets.
2. More than 10 people, research on human body activity and posture
3. Research on compatibility with unfamiliar environments.

In general, this work is to facilitate subsequent researchers who want to continue to study this research field to have a quick and comprehensive understanding of the current state-of-the-art.

In the following Table IV, a display will be made of all the models used in the technical methods of Human tracking, Huamn Activity Recognition and Human Posture Estimation and the corresponding results obtained, so as to facilitate the researchers who view this article Have an intuitive understanding of existing

technical methods, and facilitate them to create more powerful technical methods in the future that can truly enter the lives of ordinary people.

Overall, the goal of human position tracking and post-activity recognition technology using mmWave radar is to create real-time personnel protection that adapts to any indoor environment or semi-open environment. Regardless of the composition of this open or semi-open space, this makes the technology adaptable to different application scenarios, such as personnel security in office environments or factory floors, or anti-theft safe room environments in home environments with limited space and simple patients. Health detection for patients with large or small body size or rehabilitation needs, the elderly, and even the prediction and prediction of potentially dangerous activities in multi-target activity spaces such as corridors and basketball court rescue. Because millimeter-wave radar has the characteristics of not being affected by the weather environment and privacy and confidentiality, this technology has great development potential and commercial value.

Table IV. Summary of Methods for Human Activity Recognition & Human Posture Estimation

Reference	Task	Deep Learning Model	Result
[1]	HAR	CNN	95.19%
[2]	Human Tracking & Identify	Bi-LSTM	0.16 m (Tracking MPE) & 89% (Identify)
[3]	HAR	Time-distributed CNN + Bi-directional LSTM	90.47%
[4]	Human Posture Estimation	2 Channels CNN	6.18cm (Joints MAE)
[5]	HAR	5 Channels-Attribute Networks	90%
[6]	HAR	2D CNN	80%
[7]	HAR	Multi-Channel 3D CNN	92.5%

[8]	Human Posture Estimation	Seq2Seq	5cm (Joints MAE)
[9]	HAR	CNN + LSTM	97%
[10]	Human Posture Estimation	CNN	5.87cm (Joints MAE)
[11]	Human Posture Estimation	Baseline Model (CNN) + Meta-learning	6cm (Joints MAE)
[12]	Human Posture Estimation	Part Detector Model + Spatial Model	12cm (Joints MLE)
[13]	Human Tracking & Posture Estimation	Forked-ConvLSTM	32.4mm (Tracking Error) & 31mm (Joints Error)
[14]	Human Posture Estimation	P4Transformer	10cm (Joints Mean Error)
[15]	HAR	3D-CNN + LSTM// 3D-CNN	97.26%
[16]	HAR	Dual-View CNN (DVCNN) model	97.61%

[Attention: "HAR" : Human Activity Recognition ;
"MPE" : Mean Position Error ;
"MLE" : Mean Localization Error ;]

Bibliography

[1] R. Zhang and S. Cao, "Real-Time Human Motion Behavior Detection via CNN Using mmWave Radar," in IEEE Sensors Letters, vol. 3, no. 2, pp. 1-4, Feb. 2019, Art no. 3500104, doi: 10.1109/LSENS.2018.2889060.

[2] P. Zhao et al., "mID: Tracking and Identifying People with Millimeter Wave Radar," 2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), Santorini, Greece, 2019, pp. 33-40, doi: 10.1109/DCOSS.2019.00028.

[3] Akash Deep Singh, Sandeep Singh Sandha, Luis Garcia, and Mani Srivastava. 2019. "RadHAR: Human Activity Recognition from Point Clouds Generated through a Millimeter-wave Radar." In Proceedings of the 3rd ACM Workshop on Millimeter-wave Networks and Sensing Systems (mmNets'19). Association for Computing Machinery, New York, NY, USA, 51–56. https://doi.org/10.1145/3349624.3356768

[4] Arindam Sengupta, Feng Jin, Renyuan Zhang, Siyang Cao: "mm-Pose: Real-Time Human Skeletal Posture Estimation using mmWave Radars and CNNs." arXiv:1911.09592

[5] Meng, Z., Fu, S., Yan, J., Liang, H., Zhou, A., Zhu, S., Ma, H., Liu, J., & Yang, N. (2020). "Gait Recognition for Co-Existing Multiple People Using Millimeter Wave Sensing." Proceedings of the AAAI Conference on Artificial Intelligence, 34(01), 849-856. https://doi.org/10.1609/aaai.v34i01.5430

[6] I. Alujaim, I. Park and Y. Kim, "Human Motion Detection Using Planar Array FMCW Radar Through 3D Point Clouds," 2020 14th European Conference on Antennas and Propagation (EuCAP), Copenhagen, Denmark, 2020, pp. 1-3, doi: 10.23919/EuCAP48036.2020.9135381.

[7] Jiang, X.; Zhang, Y.; Yang, Q.; Deng, B.; Wang, H. "Millimeter-Wave Array Radar-Based Human Gait Recognition Using Multi-Channel Three-Dimensional Convolutional Neural Network." Sensors 2020, 20, 5466. https://doi.org/10.3390/s20195466

[8] Sengupta, Arindam et al. "NLP based Skeletal Pose Estimation using mmWave Radar Point-Cloud: A Simulation Approach." 2020 IEEE Radar Conference (RadarConf20) (2020): 1-6.

[9] Y. Kim, I. Alnujaim and D. Oh, "Human Activity Classification Based on Point

Clouds Measured by Millimeter Wave MIMO Radar With Deep Recurrent Neural Networks," in IEEE Sensors Journal, vol. 21, no. 12, pp. 13522-13529, 15 June15, 2021, doi: 10.1109/JSEN.2021.3068388.
[10] Sizhe An and Umit Y. Ogras. 2021. “MARS: mmWave-based Assistive Rehabilitation System for Smart Healthcare.” ACM Trans. Embed. Comput. Syst. 20, 5s, Article 72 (October 2021), 22 pages. https://doi.org/10.1145/3477003
[11] Sizhe An and Umit Y. Ogras. 2022. “Fast and scalable human pose estimation using mmWave point cloud.” In Proceedings of the 59th ACM/IEEE Design Automation Conference (DAC '22). Association for Computing Machinery, New York, NY, USA, 889–894. https://doi.org/10.1145/3489517.3530522
[12] H. Cui and N. Dahnoun, "Real-Time Short-Range Human Posture Estimation Using mmWave Radars and Neural Networks," in IEEE Sensors Journal, vol. 22, no. 1, pp. 535-543, 1 Jan.1, 2022, doi: 10.1109/JSEN.2021.3127937.

[13] Hao Kong, Xiangyu Xu, Jiadi Yu, Qilin Chen, Chenguang Ma, Yingying Chen, Yi-Chao Chen, and Linghe Kong. 2022. “M3Track: mmwave-based multi-user 3D posture tracking.” In Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services (MobiSys '22). Association for Computing Machinery, New York, NY, USA, 491–503. https://doi.org/10.1145/3498361.3538926
[14] Anjun Chen, Xiangyu Wang, Shaohao Zhu, Yanxu Li, Jiming Chen, and Qi Ye. 2022. “MmBody Benchmark: 3D Body Reconstruction Dataset and Analysis for Millimeter Wave Radar.” In Proceedings of the 30th ACM International Conference on Multimedia (MM '22). Association for Computing Machinery, New York, NY, USA, 3501–3510. https://doi.org/10.1145/3503161.3548262
[15] Huang, Y.; Li, W.; Dou, Z.; Zou, W.; Zhang, A.; Li, Z. “Activity Recognition Based on Millimeter-Wave Radar by Fusing Point Cloud and Range–Doppler Information.” Signals 2022, 3, 266-283. https://doi.org/10.3390/signals3020017
[16] C. Yu, Z. Xu, K. Yan, Y. -R. Chien, S. -H. Fang and H. -C. Wu, "Noninvasive Human Activity Recognition Using Millimeter-Wave Radar," in IEEE Systems Journal, vol. 16, no. 2, pp. 3036-3047, June 2022, doi: 10.1109/JSYST.2022.3140546.
[17] Li Z, Ni H, He Y, et al. “mmBehavior: Human Activity Recognition System of millimeter-wave Radar Point Clouds Based on Deep Recurrent Neural Network.” Research Square; 2023. DOI: 10.21203/rs.3.rs-2615448/v1.
[18] Lee, G.; Kim, J. Improving “Human Activity Recognition for Sparse Radar Point Clouds: A Graph Neural Network Model with Pre-Trained 3D Human-Joint Coordinates.” Appl. Sci. 2022, 12, 2168. https://doi.org/10.3390/app12042168
[19] F. Jin et al., "Multiple Patients Behavior Detection in Real-time using mmWave Radar and Deep CNNs," 2019 IEEE Radar Conference (RadarConf), Boston, MA, USA, 2019, pp. 1-6, doi: 10.1109/RADAR.2019.8835656.
[20] F. Jin, A. Sengupta and S. Cao, "mmFall: Fall Detection Using 4-D mmWave

Radar and a Hybrid Variational RNN AutoEncoder," in IEEE Transactions on Automation Science and Engineering, vol. 19, no. 2, pp. 1245-1257, April 2022, doi: 10.1109/TASE.2020.3042158.
[21] Mingmin Zhao, Yonglong Tian, Hang Zhao, Mohammad Abu Alsheikh, Tianhong Li, Rumen Hristov, Zachary Kabelac, Dina Katabi, and Antonio Torralba. 2018. “RF-based 3D skeletons.” In Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication (SIGCOMM '18). Association for Computing Machinery, New York, NY, USA, 267–281. https://doi.org/10.1145/3230543.3230579
[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun : “Deep Residual Learning for Image Recognition.” arXiv:1512.03385
[23] Qi, C. R.; Su, H.; Mo, K.; and Guibas, L. J. 2017a. “Pointnet: Deep learning on point sets for 3d classification and segmentation.” In 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017, 77–85.
[24] Qi, C. R.; Yi, L.; Su, H.; and Guibas, L. J. 2017b. “Pointnet++: Deep hierarchical feature learning on point sets in a metric space.” In Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA, 5099–5108.
[25] Ke Sun, Bin Xiao, Dong Liu, Jingdong Wang : “Deep High-Resolution Representation Learning for Human Pose Estimation.” 25 Feb 2019. arXiv:1902.09212
[26] Pearce, A.; Zhang, J.A.; Xu, R.; Wu, K. “Multi-Object Tracking with mmWave Radar: A Review.” Electronics 2023, 12, 308. https://doi.org/10.3390/electronics12020308
[27] F.J.; Zhang, Y.; Fu, M.; Li, Y.; Deng, Z. “Application of Deep Learning on Millimeter-Wave Radar Signals: A Review.” Sensors 2021, 21, 1951. https://doi.org/10.3390/s21061951
[28] Sapiezynski P, Stopczynski A, Gatej R, Lehmann S (2015) “Tracking Human Mobility Using WiFi Signals.” PLoS ONE 10(7): e0130824. https://doi.org/10.1371/journal.pone.0130824
[29] Choi, J.W.; Nam, S.S.; Cho, S.H. “Multi-Human Detection Algorithm Based on an Impulse Radio Ultra-Wideband Radar System.” IEEE Access 2016, 4, 10300–10309. [CrossRef]
[30] Wenjun Jiang, Chenglin Miao, Fenglong Ma, Shuochao Yao, Yaqing Wang, Ye Yuan, Hongfei Xue, Chen Song, Xin Ma, Dimitrios Koutsonikolas, Wenyao Xu, and Lu Su. 2018. “Towards Environment Independent Device Free Human Activity Recognition.” In Proceedings of the 24th Annual International Conference on Mobile Computing and Networking (MobiCom '18). Association for Computing Machinery, New York, NY, USA, 289–304. https://doi.org/10.1145/3241539.3241548
[31] Chiani, M.; Giorgetti, A.; Paolini, E. “Sensor Radar for Object Tracking.” Proc. IEEE 2018, 106, 1022–1041. [CrossRef]
[32] Z. Yan, T. Duckett and N. Bellotto, "Online learning for human classification

in 3D LiDAR-based tracking," 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 2017, pp. 864-871, doi: 10.1109/IROS.2017.8202247.
[33] Kun Qian, Zhaoyuan He, and Xinyu Zhang. 2020. “3D Point Cloud Generation with Millimeter-Wave Radar.” Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 4, 4, Article 148 (December 2020), 23 pages. https://doi.org/10.1145/3432221
[34] Xian-Feng Han, Jesse S. Jin, Ming-Jie Wang, Wei Jiang, Lei Gao, Liping Xiao, “A review of algorithms for filtering the 3D point cloud.” Signal Processing: Image Communication, Volume 57, 2017, Pages 103-112, ISSN 0923-5965, https://doi.org/10.1016/j.image.2017.05.009.
[35] R.B. Rusu, S. Cousins, 3d is here: Point cloud library (pcl), in: IEEE International Conference on Robotics and Automation, 2011, pp. 1–4.
[36] Aldoma A., Marton Z.C., Tombari F., Wohlkinger W., Potthast C., Zeisl B., et al. Tutorial: Point cloud library: Three-dimensional object recognition and 6 dof pose estimation. IEEE Robot. Autom. Mag., 19 (3) (2012), pp. 80-91
[37] Saval-Calvo M., Orts-Escolano S., Azorin-Lopez J., Garcia-Rodriguez J., Fuster-Guillo A., Morell-Gimenez V., et al. A comparative study of downsampling techniques for non-rigid point set registration using color Bioinspired Computation in Artificial Systems, Springer (2015), pp. 281-290
[38] Pfister H., Gross M. Point-based computer graphics IEEE Comput. Graph. Appl., 24 (4) (2004), pp. 22-23
[39] Kobbelt L., Botsch M. A survey of point-based techniques in computer graphics Comput. Graph., 28 (6) (2004), pp. 801-814
[40] D. Holz, S. Behnke, Fast range image segmentation and smoothing using approximate surface reconstruction and region growing, in: International Conference on Intelligent Autonomous Systems, 2012, pp. 61–73.
[41] Han J., Shao L., Xu D., Shotton J. Enhanced computer vision with microsoft kinect sensor: A review IEEE Trans. Cybernet., 43 (5) (2013), pp. 1318-1334
[42] Xu W., Lee I.S., Lee S.K., Lu B., Lee E.J. Multiview-based hand posture recognition method based on point cloud Ksii Trans. Internet Inf. Syst., 9 (7) (2015), pp. 2585-2598
[43] B. Huhle, T. Schairer, P. Jenke, W. Strasser, Robust non-local denoising of colored depth data, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Anchorge, AK, June, 2008, pp. 1–7.
[44] Landa J., Procházka D., Šťastný J. Point cloud processing for smart systems Acta Univ. Agricult. Silvicult. Mendelianae Brunensis, 61 (7) (2013), pp. 2415-2421
[45] Xie H., McDonnell K.T., Qin H. Surface Reconstruction of Noisy and Defective Data Sets, Visualization, IEEE, Austin, TX, USA (Oct., 2004), pp. 259-266
[46] E.A.L. Narváez, N.E.L. Narváez, Point cloud denoising using robust principal component analysis, in: Proceedings of the First International Conference on Computer Graphics Theory and Applications, Setúbal, Portugal, February, 2006, pp. 51–58.

[47] F. Zaman, Y.P. Wong, B.Y. Ng, Density-based denoising of point cloud, 2016. ArXiv preprint arXiv:160205312.
[48] J. Park, H. Kim, Y.W. Tai, M.S. Brown, I. Kweon, High quality depth map up-sampling for 3d-tof cameras, in: International Conference on Computer Vision, Barcelona, Nov., 2011, pp. 1623–1630.
[49] Bilik, I.; Longman, O.; Villeval, S.; Tabrikian, J. The rise of radar for autonomous vehicles: Signal processing solutions and future research directions. Ieee Signal Process. Mag. 2019, 36, 20–31. [CrossRef]
[50] Richards, M.A. Fundamentals of Radar Signal Processing. McGraw-Hill Education: Georgia Institute of Technology, Atlanta, GA, USA, 2014.
[51] Iovescu, C.; Rao, S. The Fundamentals of Millimeter Wave Sensors. Texas Instruments Inc.: Dallas, TX, USA, 2017; pp. 1–8.
[52] Ramasubramanian, K.; Ramaiah, K. Moving from legacy 24 ghz to state-of-the-art 77-ghz radar. Atzelektronik Worldw. 2018, 13, 46–49. [CrossRef]
[53] Pelletier, M.; Sivagnanam, S.; Lamontagne, P. Angle-of-arrival estimation for a rotating digital beamforming radar. In Proceedings of the 2013 IEEE Radar Conference (RadarCon13), Ottawa, ON, Canada, 29 April–3 May 2013; pp. 1–6.
[54] Stoica, P.; Wang, Z.; Li, J. Extended derivations of MUSIC in the presence of steering vector errors. IEEE Trans. Signal Process. 2005, 53, 1209–1211. [CrossRef]
[55] Rohling, H.; Moller, C. Radar waveform for automotive radar systems and applications. In Proceedings of the 2008 IEEE Radar Conference, Rome, Italy, 26–30 May 2008; pp. 1–4.
[56] Kuroda, H.; Nakamura, M.; Takano, K.; Kondoh, H. Fully-MMIC 76 GHz radar for ACC. In Proceedings of the ITSC2000. 2000 IEEE Intelligent Transportation Systems. Proceedings (Cat. No. 00TH8493), Dearborn, MI, USA, 1–3 October 2000; pp. 299–304.
[57] Wang, W.; Du, J.; Gao, J. Multi-target detection method based on variable carrier frequency chirp sequence. Sensors 2018, 18, 3386. [CrossRef]
[58] Song, Y.K.; Liu, Y.J.; Song, Z.L. The design and implementation of automotive radar system based on MFSK waveform. In Proceedings of the E3SWeb of Conferences, 2018 4th International Conference on Energy Materials and Environment Engineering (ICEMEE 2018), Zhuhai, China, 4 June 2018; Volume 38, p. 01049.

[59] Marc-Michael, M. Combination of LFCM and FSK modulation principles for automotive radar systems. In Proceedings of the German Radar Symposium, Berlin, Germany, 11–12 October 2000.
[60] Gao, X.; Xing, G.; Roy, S.; Liu, H. Experiments with mmwave automotive radar test-bed. In Proceedings of the 2019 53rd Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, 3–6 November 2019; pp. 1–6.
[61] Finn, H. Adaptive detection mode with threshold control as a function of spatially sampled clutter-level estimates. Rca Rev. 1968, 29, 414–465.
[62] Hezarkhani, A.; Kashaninia, A. Performance analysis of a CA-CFAR detector

in the interfering target and homogeneous background. In Proceedings of the 2011 International Conference on Electronics, Communications and Control (ICECC), Ningbo, China, 9–11 September 2011; pp. 1568–1572.
[63] Messali, Z.; Soltani, F.; Sahmoudi, M. Robust radar detection of CA, GO and SO CFAR in Pearson measurements based on a non linear compression procedure for clutter reduction. SignalImage Video Process. 2008, 2, 169–176. [CrossRef]
[64] Sander, J.; Ester, M.; Kriegel, H.-P.; Xu, X. Density-based clustering in spatial databases: The algorithm gdbscan and its applications. Data Min. Knowl. Discov. 1998, 2, 169–194. [CrossRef]
[65] Texas Instruments. 2019. IWR1443 single-chip 76-GHz to 81-GHz mmWave sensor evaluation module IWR1443BOOST (ACTIVE). http://www.ti.com/tool/IWR1443BOOST Accessed: 2019-07-05.
[66] Texas Instruments. 2018. IWR1443BOOST Evaluation Module User‘s Guide. http://www.ti.com/lit/ug/swru518c/swru518c.pdf (2018). Accessed: 2019-07-05.
[67] Tianwei Xing, Sandeep Singh Sandha, Bharathan Balaji, Supriyo Chakraborty, and Mani Srivastava. 2018. Enabling Edge Devices that Learn from Each Other: Cross Modal Training for Activity Recognition. In Proceedings of the 1st International Workshop on Edge Systems, Analytics and Networking. ACM, 37–42.
[68] C. Schuldt, I. Laptev and B. Caputo, "Recognizing human actions: a local SVM approach," Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004., Cambridge, UK, 2004, pp. 32-36 Vol.3, doi: 10.1109/ICPR.2004.1334462.
[69] Taud, H., Mas, J. (2018). Multilayer Perceptron (MLP). In: Camacho Olmedo, M., Paegelow, M., Mas, JF., Escobar, F. (eds) Geomatic Approaches for Modeling Land Change Scenarios. Lecture Notes in Geoinformation and Cartography. Springer, Cham. https://doi.org/10.1007/978-3-319-60801-3_27
[70] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," in Neural Computation, vol. 9, no. 8, pp. 1735-1780, 15 Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
[71] Seung-Hwan Bae, Kuk-Jin Yoon; Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 1218-1225
[72] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, “Pointnet: Deep learning on point sets for 3d classification and segmentation,” Proc. Computer Vision and Pattern Recognition (CVPR), IEEE, vol. 1, no. 2, p. 4, 2017.
[73] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," in Neural Computation, vol. 9, no. 8, pp. 1735-1780, 15 Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
[74] Fei, H.; Tan, F. Bidirectional Grid Long Short-Term Memory (BiGridLSTM): A Method to Address Context-Sensitivity and Vanishing Gradient. Algorithms 2018, 11, 172. https://doi.org/10.3390/a11110172
[75] RF-Based 3D Skeletons.Mingmin Zhao, Yonglong Tian, Hang Zhao,

Mohammad Abu Alsheikh, Tianhong Li, Rumen Hristov, Zachary Kabelac, Dina Katabi, Antonio Torralba Massachusetts Institute of Technology
[76] Makihara, Y.; Suzuki, A.; Muramatsu, D.; Li, X.; and Yagi, Y. 2017. Joint intensity and spatial metric learning for robust gait recognition. In 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21- 26, 2017, 6786–6796.
[77] Li, S.; Liu, W.; Ma, H.; and Zhu, S. 2018. Beyond view transformation: Cycle-consistent global and partial perception gan for view-invariant gait recognition. In 2018 IEEE International Conference on Multimedia and Expo, ICME 2018, San Diego, CA, USA, July 23-27, 2018, 1–6.
[78] Chao, H.; He, Y.; Zhang, J.; and Feng, J. 2019. Gaitset: Regarding gait as a set for cross-view gait recognition. In The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019., 8126– 8133.
[79] Li, S.; Liu, W.; and Ma, H. 2019. Attentive spatial-temporal summary networks for feature learning in irregular gait recognition. IEEE Trans. Multimedia 21(9):2361–2375.
[80] Wang, W.; Liu, A. X.; and Shahzad, M. 2016. Gait recognition using wifi signals. In Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, 363–373.
[81] Zeng, Y.; Pathak, P. H.; and Mohapatra, P. 2016. Wiwho: wifibased person identification in smart spaces. In Proceedings of the 15th International Conference on Information Processing in Sensor Networks, 4. IEEE Press
[82] Zou, H.; Zhou, Y.; Yang, J.; Gu, W.; Xie, L.; and Spanos, C. J. 2018. Wifi-based human identification via convex tensor shapelet learning. In Thirty-Second AAAI Conference on Artificial Intelligence.
[83] Qi, C. R.; Su, H.; Mo, K.; and Guibas, L. J. 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation. In 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017, 77–85.
[84] IWR1843 Single-Chip 76- to 81-GHz FMCW mmWave Sensor (Rev. A). Available online: https://www.ti.com.cn/documentviewer/ cn/IWR1843/datasheet/GUID-4FBB6021-CAC6-45C5-B361-E49F964BFB22#TI TLE-SWRS188X2481 (accessed on 9 February 2022).
[85] J. A. Oulton, “The global nursing shortage: an overview of issues and actions,” Policy, Politics, & Nursing Practice, vol. 7, no. 3 suppl, pp. 34S–39S, 2006.
[86] C.-W. Lee, Y.-S. Wang, T.-Y. Hsu, K.-Y. Chen, H.-y. Lee, and L.- s. Lee, “Scalable sentiment for sequence-to-sequence chatbot response with performance analysis,” in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6164–6168, IEEE, 2018.
[87] K. Yu, H. Li, and B. Oguz, “Multilingual seq2seq training with similarity loss

for cross-lingual document classification," in Proceedings of The Third Workshop on Representation Learning for NLP, pp. 175–179, 2018.
[88] Y. Zhang and W. Xiao, "Keyphrase generation based on deep seq2seq model," IEEE Access, vol. 6, pp. 46047–46057, 2018
[89] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Works), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
[90] Radford, Alec, Narasimhan, Karthik, Salimans, Tim and Sutskever, Ilya. "Improving language understanding by generative pre-training." (2018): .
[91] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language Models are Unsupervised Multitask Learners.
[92] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS'20). Curran Associates Inc., Red Hook, NY, USA, Article 159, 1877–1901.
[93] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17). Curran Associates Inc., Red Hook, NY, USA, 6000–6010.
[94] MAML-Pytorch https://github.com/dragen1860/MAML-Pytorch
[95]Natural_language_processing.https://en.wikipedia.org/wiki/Natural_language_processing
[96] Short-Range Radar - A Look behind the Scenes. url: https://www.ilmsens.com/short-range-radar/ (cit. on p. 32).
[97] Radar a onda continua modulata in frequenza. url: https://www.radartutorial.eu/02.basics/rp08.it.html (cit. on p. 33).
[98] Introduction to the Principle of Frequency Modulated Continuous Wave FMCW Radar. https://cdsentec.com/introduction-to-the-principle-of-frequency-modulated-continuous-wave-fmcw-radar/#:~:text=At%20the%20same%20time%20when%20FMCW%20radar%20transmits,thereby%20reducing%20the%20probability%20of%20interception%20and%20interference.
[99] RadHAR: Human Activity Recognition from Point Clouds Generated through a Millimeter-wave Radar. https://github.com/nesl/RadHAR

[100] IWR1642 Evaluation Module (IWR1642BOOST) Single Chipmm Wave Sensing Solution
[101] IWR6843 Single-chip 60-GHz to 64-GHz intelligent mmWave sensor integrating processing capability. https://www.ti.com/product/IWR6843
[102] AWR1443, AWR1243E valuation Module (AWR1443BOOST, AWR1243BOOST) mmWave Sensing Solution User's Guide.
[103] IWR6843ISK-ODS IWR6843 intelligent mmWave overhead detection sensor (ODS) antenna plug-in module. https://www.ti.com/tool/IWR6843ISK-ODS