Techniques to Process point cloud data to perform HAR using Machine learning

AUTHOR, Mian Tashfeen Shahid Anwar , s2039761 , University of Twente, The Netherlands

ABSTRACT. Human Activity Recognition (HAR) can solve a lot of problems for example it can be used to monitor elderly people in a better way and even in security purposes or to track movements in rooms. To perform Human Activity Recognition data in is collected in form of a point cloud using a (mmWave) millimeter wave radar. The problem we want to solve is that how can we do it effectively and accurately (accuracy , data , speed) which is feasible for most use cases. In this paper first of all all methods for signal pre processing and Machine Learning (ML) were considered and then compared to shortlist the most important and effective ones and try to implement and test them. Our results are based on the accuracy achieved by the ML methods. We propose a new signal pre processing method which is a combination of frame divider , dbscan , rotations , enhanced voxelization and voxelization.

*Keywords.*Point cloud , Signal Processing ,mmWave , CNN , LSTM , FCMW , Voxelization , DVCNN , BI Directional LSTM.

1 INTRODUCTION

Human Activity Recognition (HAR) can be a very integral part of our lives in the coming future. The main reason for this is that it can assist us in our day to day lives in a lot of ways. For example We can monitor elderly people using HAR it can be used in hospitals[10]. Moreover one of the most important issues which can be solved using HAR is monitoring of elderly people. HAR has been explored by different kinds of devices, including cellular phones, wireless Internet, wearable sensor devices, visible-light cameras , infrared cameras , LIDAR imaging devices and indoor Doppler radar. Different signal wave forms are acquired from the aforementioned devices, which can then be pre processed and later we can provide them as input to the Machine Learning (ML) classifiers. The fact that we need better , easier and accurate methods of HAR has a issue that we need to improve and find new ML algorithms and signal/data pre processing methods for HAR. In this research different ML classifiers and signal processing algorithms were explored. First step was collection of data. (FMCW mmwave radar) millimeter wave radar is used because radars do not capture ambient information which other methods of data collection do such as cameras [4][2]. The use of this radar is that we can install it in a room or any other place and it has a sensing range. The radar senses the activity being taken place and the output we get from the radar is in the form of spectrograms or point cloud data.

TScIT 37, July 8, 2022, Enschede, The Netherlands

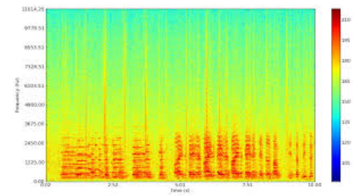

Fig. 1. Representation of Spectrogram

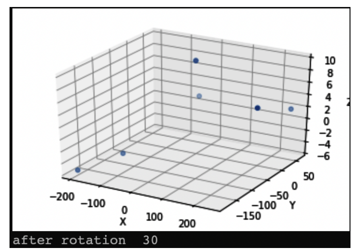

Fig. 2.
Representation of Point cloud

Fig 1 is a representation of spectrogram. We can notice that the darker colours represent intensity but it is not represented clearly using points. Its just tells you intensity of the activity. Research was carried about both of these forms of data with the reference to how they are being used in HAR. As the research suggested using point cloud was a better idea than using spectrogram because if point cloud is used we can make changes and alter it in many ways. On the other hand methods to alter the spectrogram were a bit harder in comparison to point clouds.

Fig 2 is a representation of a point cloud captures by a millimeter wave radar. We can see that it is a three dimensional cube where the points are representing the activity being taken place. Moreover most literature which was reviewed used point clouds instead of spectrogram but there were some projects which used spectrogram as well (and they have a good accuracy as well). We opted for the point cloud method.The reasons for this are that point clouds are easy to analyse during the development phase. Changes can be visualized and seen directly from point clouds. After this we are left with one more step before we can successfully perform HAR. First data needs to be pre processed and transformed into a form which can be fed into a ML classifier. The goal is to get as much high accuracy as possible
. We use frame divider [9] , rotations[12] , DBSCAN[12] , enhanced voxelization [12] and lastly voxelization [4].

2 SIGNAL / DATA PRE PROCESSING

There exist a lot of techniques to preprocess the point clouds before feeding them into the classifier. While reviewing the literature it was noticed that different algorithms were being used for this task. The most prominent and effective ones include Frame Divider helps reducing the size of data by clubbing together points from different frames from pantomime [9]) , DBScan [11] removes noisy points from the point cloud from DVCNN[12] , pantomime[9] , Rotations Used to enhance the data set by rotating the point cloud by different angles and produces new point clouds from DVCNN[12] , Enhanced Voxelization Used to make the point cloud into fixed dimensions from DVCNN [12] and Voxelization[4][12] (Voxelization is one of the most important technique. It takes a cluster of points and represent those points using cubes containing that cluster from radhar[4] , DVCNN[12] , pantomime[9]

3 MACHINE LEARNING / DEEP LEARNING CLASSIFIERS

Machine Learning (ML) [12][4][9]can be used to know a lot about humans and animals. However Its integration in smart spaces is still in its early stages. At this time we have radars (mm Wave[4]) and we can collect all sorts of data(Using frequencies , vibrations and electromagnetic waves)[4] Importance of ML and signal processing for activity recognition using FMCW[4] mm wave radar can be realized from the fact that ML is the core concept which lies beneath all the data we are collecting and extracting different features so that we can feed it into the neural network and train it. Lastly use it in different ways for classification and other purposes. Adopted techniques include neural network architectures which basically analyse data (about fitness , activity recognition) using point clouds and spectrograms[5] generated by FMCW[4] mm wave radar by processing that data using techniques like (voxelization , enhanced voxelization , rotations , DBSCAN[11] , frame divider[9]). Our data sets were tested on three ML classifiers so far. SVM , Bi LSTM , Time Distributed CNN + Bi Directional LSTM[4][12].

4 PROBLEM DEFINITION

If we take table 1.1 into consideration We can see that most algorithms have accuracy of 90[4] percent at least like DVCNN [12]. We propose a new system based on best Machine Learning (ML) methods and signal pre processing methods. The problem we are trying to solve is to increase the accuracy of the classifier by using different signal pre processing steps.

5 OVERVIEW OF SIGNAL/DATA PROCESSING ALGORITHMS

5.1 Type of Data being used

As explained in the introduction that the type of radar used to collect data is a millimeter wave radar (FCMW)[4]. The output of the radar is in the form of a 3D point cloud.
The point cloud in Fig 3 contains all the points alongside their velocities and intensities. Which can be used to run different algorithms. There are five different signal pre processing algorithms which are used in Human Activity Recognition to process point clouds generated from millimeter wave radar. We explain them one by one.

5.2 Frame Divider

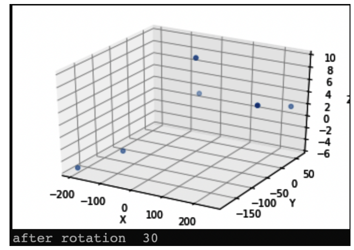

Fig. 3. Type of data

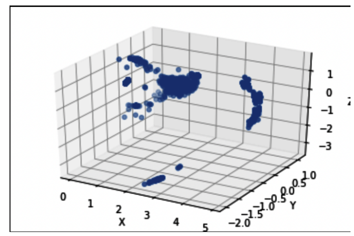

Fig. 4. Representation of frame after frame divider

This approach is adopted from pantomime[9]. What frame divider does is that it takes all the points in all the point cloud frames and represents them using less frames. For example if we have 100 frames it takes all of those 100 frames and divides the total number of points by k (where k is the number of frames we want). This means that if k is 2 then the first half of the points will be represented using frame 1 and the rest using frame 2
Fig 4 shows a point cloud after retaining all the points from all frames in a time window and then plotting them into one point cloud. There are some benefits to using this approach. First of all it reduces the data size significantly For example in pantomime paper they explain that if there are 100 frames we represent those 100 frames using only 4 to 8 frames which means that the 4 processed frames contain all the information and we do not have to process every frame separately which reduces the extra time needed and it reduces the data size from the order of GB's to the order of MB's. Time decay is applied which does not change the sequence of frames. for example if there are 5 frames and every frame has 5 points so 25 points in total. Now when we use frame divider it only determines the number of frames we want to represent these points in. So for a 2 second window lets suppose there are 5 frames now if we make it into 2 frames we are representing 1st second using the first frame and the second one using the second frame. The size of data increases or decreases based on the number of frames being used to represent the points it does not matter if we have more points or less points in a frame. Moreover [9] also makes the number of points in the frame equal because of the machine learning algorithm which they are using it requires equal number of points. Apart from this they just use the time decay

5.3 DBSCAN

One of the other most important method is DBSCAN, it is used by almost all other papers[12][11][9] except RADHAR[4]. (Regardless

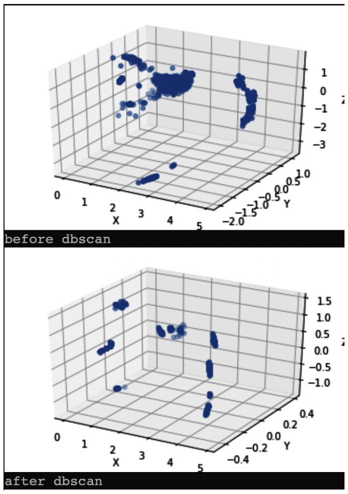

Fig. 5. Representation before and after DBSCAN

of what Machine Learning (ML) algorithm they are using.) What DBSCAN[12] does is that in every point cloud there are some noisy points which are either very low in intensity or belong to another disruption/movement in the sensing area. It takes in two parameters namely radius around the point and minimum number of points in that radius which means that if a point has at least minimum points in the radius around it only then it is marked as a centeroid point [12] (centeroid means a point which has strong ties in a cluster) after checking for all centeroids we check all the neighbours of the centeroids and see if they are centeroids and lastly we remove all the remaining points which can be classified as noisy points (non centeroid points). This helps us clean the point clouds to a certain extent which helps in making the resolution or quality of data better.
Fig 5 shows a point cloud frame before and after DBSCAN as it can be seen that it removes a lot of points from the point cloud which are selected based on the number of links do these points have with other points in its radius. We set the radius to 0.01m and minimum points to 60 because we are using frame divider[9] which increases the number of points in a frame.

5.4 Rotations

The idea of Rotations is taken from DVCNN[12] and pantomime[9]. Rotations technique is also very important its main purpose is to enhance the data set. We needed to enhance the data set because first of all in our method and other methods which achieve high accuracy everyone used DBSCAN which removes noisy points due to that it decreases the amount of information which can be extracted. Moreover if data set is rotated the system can have a view from different angles which means it improves the quality of data being provided to the Machine Learning (ML) algorithm. Apart from this frame divider is used in pantomime and our method which makes the data set very small so it is important to have more data after using frame divider and DBSCAN[9][12][11]. The data is compressed as points are clubbed together to represent a larger time window in a frame. Which basically makes our data set very small per file so that is why rotations are helpful, we can also use other symmetry properties and change the dataset. It does not makes the algorithm decide incorrectly because for different activities the general point scattering is different. For example when we remove the noisy points from the boxing file it will give us mostly the same pattern as compared to the walk or squats. So when you rotate these point clouds they still represent a different pattern for each activity, which in return also makes the classification better. For performing rotations all points are considered in a frame then First they are converted into polar coordinates and after that rotated using the respective formulas Fig 6 shows the equations to convert the points into cartesian points and Fig 7 shows the equations for converting the cartesian points into normal rotated points. Fig 8 is a representation of the frame before and after rotation of 90 degrees
Fig 9 shows the representation of the xy plane as well. we can see that how the whole point cloud is being shifted to give us a view from another angle.

$$r = \sqrt{x^2 + y^2 + z^2}$$
$$\theta = \cos^{-1}(z/r)$$
$$\phi = \tan^{-1}(y/x).$$

Fig. 6. Formulas for converting a point into Cartesian frame

$$x = r\sin(\theta)\cos(\phi + \delta\phi) + \delta x$$
$$y = r\sin(\theta)\sin(\phi + \delta\phi) + \delta y$$
$$z = r\cos(\theta).$$

Fig. 7. Formulas to convert the Cartesian points to normal rotated points.

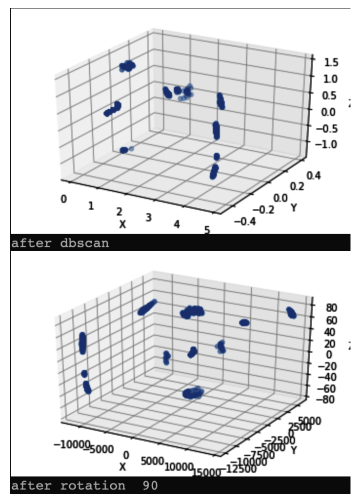

Fig. 8. Before and after rotation

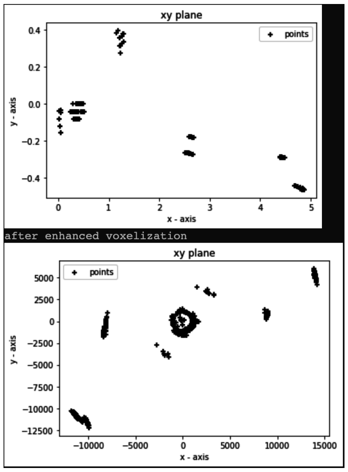

Fig. 9. XY view Before and after rotation

5.5 Enhanced voxelization

Enhanced voxelization is picked from the DVCNN[12] paper. it is also very useful and important because it helps transform the point cloud into a better resolution. Enhanced voxelization is very simple it only means that the point cloud window for every frame have fixed dimensions. In previous methods there is no limit to the point cloud window but here we set the boundaries of the point cloud window and we remove all the points which lie outside that boundary and made the point cloud fixed in terms of points scattering. Moreover we do this step before voxelizing[12][4] the point cloud.

5.6 Voxelization

The last step performed in signal processing is voxelization Fig 10 shows us the representation of conventional voxelization method against our method. there are no changes to the code we use the conventional method from RADHAR[4] this is because previously the problem was with the quality of the point cloud being voxelized contained a lot of noisy points and it was also not set to boundaries or rotated. Our method gives in a very precise point cloud which contains everything very clearly so that we get less voxels as we were getting previously because of noisy points and unscattered points. Moreover voxelization is the most important part of the whole system because we cannot perform Human Activity Recognition (HAR) using only points. On the other hand voxelization is used by all methods who use a point clouds.

6 OVERVIEW OF ML/DL ALGORITHMS

There are different types of Machine Learning (ML) algorithms used overall ranging from SVM classifier to dual view CNN. First of all we are going to explain about how neural networks work. There are three type of neural networks. ANN|(Artificial neural network)(Processes information in only forward direction) , CNN and RNN(Recurrent Neural network ((RNN) are more complex. They save the output of processing nodes and feed the result back into the model (they do not pass the information in one direction only). This is how the model is said to learn to predict the outcome of a layer. Each node in the RNN model acts as a memory cell, continuing the computation and implementation of operations. If the network's prediction is incorrect, then the system self-learns and continues working towards the correct prediction during back propagation.)

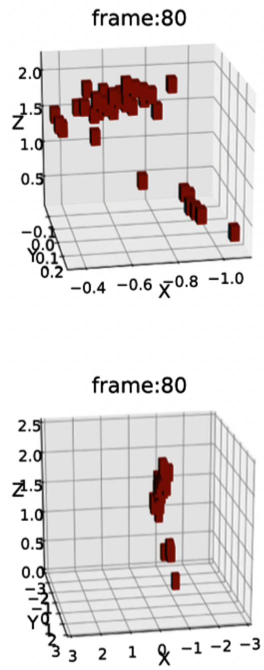

Fig. 10. Voxelization on conventional method vs our method

6.1 SVM

SVM [4][12][11] is used for linearly separable data, which means if a data set can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier. SVM needs more data to train in terms of frames generally however DVCNN and our method uses some algorithms which reduces the amount of data hence SVM is not able to perform very well with our processed data however our method achieves better accuracy than DVCNN this is mainly because we use frame divider which increases the amount of points per frame.

6.2 BILSTM

Bidirectional long short term memory(BILSTM)[4][12][11][9] is the process of making any neural network to have the sequence information in both directions backwards (future to past) or forward(past to future). In bidirectional, our input flows in two directions, making a BILSTM different from the regular LSTM. LSTM is a recurrent neural network which feeds back the information. This algorithm does not work with time windows so it takes a frame learns from it and repeat for other frames and then again.

6.3 TD CNN+BIlSTM

CNN (Convolution Neural Network)[4][11][12] is a type of feed-forward artificial network where the connectivity pattern between its neurons is inspired by the organization of the animal visual cortex. The visual cortex has a small region of cells that are sensitive to specific regions of the visual field. Moreover Bi directional LSTM is explained above. time distributed cnn plus BILSTM has a time window and we use a technique of frame divider which basically combines the time windows which means that now more time sensitive information is represented by less frames. Moreover this neural network also needs very clear data which we achieve by pre processing those point clouds and enhancing the data set as well using rotations.

7 OUR METHOD

The goal of our method is to only pre process data, then feed into the existing classifiers and see if we can improve the accuracy of those classifiers by pre processing the data and transforming it into a better form. We take inspiration from the RADHAR[4] , DVCNN[12] , Pantomime[9] and Harnet[11]. From RADHAR[4] we use and improve their signal pre processing code and classifiers in terms of accuracy. For DVCNN[12] we adopt their data pre processing techniques (DBSCAN Rotations enhanced voxelization)[9][4][11][12]. For pantomime[9] we adopt their data pre processing methods (rotations , frame divider).

7.1 OUR SIGNAL PROCESSING

Our signal pre processing is based on five steps.First of all frame division is performed which transforms a large number of frames into very less frames retaining all the important information after that DBSCAN is performed which removes all the noisy and unwanted points. Next rotations which is a data enhancing technique we take the point cloud frame which we get after frame division and DBSCAN and then perform rotations on that (rotations can be performed at any angle and it gets a new frame every time). After this the last step is voxelization for voxelization conventional voxelization method is used from radhar and we do a small change called enhanced voxelization before voxelizing the frame that we make the dimensions of the cube to be 6x6x2.5 meters as explained in DVCNN.

7.2 Our ML/DL method

For our Machine Learning (ML) classifier we propose a method which we think can achieve a very good accuracy. But we did not compile it into a classifier yet [12][5].
We think that our data will work well if it is run on a ML classifier like DVCNN where they use a dual view approach (They convert the 3d point cloud into 2d frames xy and yz frames. However our method proposes to take xy , yz and xz frames and then run a time distributed cnn layer on these three frames and then we join the frames together again into a point cloud and run a BILSTM layer because our method achieves a very good accuracy on BILSTM. Lastly we want to combine learning in these two somehow. For example we want to use only the time distribution part from the time distributed cnn and then apply that to a LSTM classifier.

8 RESULTS

This method was tested on[4][12] SVM , BILSTM and TD CNN+BILSTM. For all of them a better accuracy was achieved than radhar and DVCNN except for the SVM classifier. The results are summarised and explained better below. model parameters are the same as radhar[4] for all the classifiers. It is tested on SVM , bi directional LSTM and tdCNN+BILSTM. The only change is the number of Epochs for TDCNN+BILSTM from 30 to 35.

Apart from this pre processed data contains the original pre processed point cloud frames alongside rotated point cloud frames as well. Frames were rotated twice once for 45 degrees and then for 90 degrees however it as also possible to enhance the data set even more. For example rotations of 30 degree intervals can be performed and rotated 12 times to get a better data set which contains more information. However it might over fit the classifiers. Alongside that for DBSCAN radius is 0.01 meters however it can also be reduced even more and the minimum points in radius for a point to be classified as a centeroid point as 60 points (This number is high as there is a high number of points in a frame because frame divider is used) This can also be increased further. SVM is tested by radhar[4] DVCNN[12] and our method. 63.75 , 25.00 , 35.00 are the accuracy's achieved by these three methods. BiLSTM is tested by RADHAR[4] , Pantomime[9] , DVCNN[12] and our method. 88.42 , 93.89 , 62.53 , 95.00 are the respective accuracy's achieved. Our BILSTM achieves a better accuracy. TD CNN+BiLSTM is tested by RADHAR , DVCNN and our method. 90.47 , 66.79 , 92.00 are the respective accuracy's achieves our method gets a good accuracy.

Method	Signal processing steps	Classifiers / Accuracy
RadHar	Voxelization	SVM=63.74 , MLP = 80.34 , BiLSTM=88.42, CnnBiLSTM=90.47
HarNet	Voxelization + DBSCAN	CnnBiGRU=91.52
Pantomime	Voxelization + DBSCAN + Rotations + Frame Divider	BiLSTM =93.89
DVCNN	Voxelization + DBSCAN + Rotations + Enhanced Voxelization	SVM=25.00 , MLP = 38.33 , BiLSTM=62.53, CnnBiLSTM=66.79, DVCNN = 97.00
Our Method	Voxelization + DBSCAN + Rotations + Enhanced Voxelization + Frame Divider	SVM=35.00 , BiLSTM=95.00, CnnBiLSTM=92.00

Table 1.1

9 DISCUSSION

Table 1.1 shows the accuracy's using different pre processing steps and different classifiers.
It can be seen that there are five methods and each one uses same but better combinations of algorithms for signal pre processing (DBSCAN , voxelization , frame division , rotations and enhanced voxelization) These methods are tested on different Machine Learning (ML) algorithms (SVM , MLP , Bi LSTM , TD CNN+BiLSTM , CNN+BiGRU[9] , DVCNN[12]) Our method was tested on SVM , BiLSTM , CNN+BiLSTM. Our data pre processing technique is not tested on CNN+BiGRU , DVCNN , MLP. Moreover our method is closely related to the method in DVCNN and pantomime pre processing technique with some changes For Example frame divider. Our method achieves better accuracy than DVCNN pre processing technique in SVM , BiLSTM and CNN + BiLSTM
There are some important points.
First of all the data at first was in the order of magnitude of GB's and it was transformed into the order of MB's because it was not being compressed when frame divider[9] is used it automatically compresses the data (because time decay[9] is applied which makes it variable), which reduces the number of frames while retaining the rest of the information
How to make data pre processing algorithms better? After that it was noticed that in the Radhar[4] paper all the points also contain the intensity of the point alongside its velocity which can be used in making the existing algorithms better for example we can make the DBSCAN algorithm even better if we also filter out points based on intensity which will tell us automatically about the noisy points and help in filter them out better. Lastly a big part of the focus was on making all of these methods as a variable which can be controlled and checked and tested on different configurations. For example variables can be controlled. The variables of DBSCAN (radius around a point and minimum points) alongside that rotations settings can be altered because you can give in an array of different angles. Moreover it is also possible to change the enhanced voxelization settings.
This algorithm is not yet completely tested on different ML classifiers and even there are a lot of more things which can be controlled now like the DBSCAN , frame divider (which directly also makes effect on time distributed frames) so there are still more things which can be done for example running all these algorithms in contrast to 2d frames (xy , yz , xz)[12][9] on 3 different layers of CNN which are pre processed from all angles alongside time distributed windows. Then we join them and bring all those windows into a 3D point cloud and run on BiLSTM layer.

10 CONCLUSION

In conclusion to the research question (Techniques to Process POINT CLOUD data to perform Human Activity Recognition (HAR) using Machine Learning (ML).) There are a lot of ways in which point cloud can be pre processed and in return good accuracy can be achieved on ML algorithms. Only by using a good configuration of algorithms while pre processing the point cloud. Also of course making new and better neural networks classifiers is possible. Overall conclusion is that the quality of signal pre processing makes a lot of effect on the accuracy of the ML classifier. There are ways this data can be used more accurate HAR. For example running Time distributed CNN on 2D frames and then combining the learning to create 3D point clouds running BiLSTM. One important thing is that time distributed CNN layers for 2d frames can be used to learn underlying features in the point cloud better and then by using those features we can get a better view of the point cloud and generate our 3d point clouds ourselves and run them on BiLSTM layer.

REFERENCES

[1] Aggarwal, J. K., Xia, L. (2014)*Human activity recognition from 3D data: A review. Pattern Recognition Letters, 48, 70–80.* https://doi.org/10.1007/978-3-030-66840-2$_6$0

[2] Zhang, H., Reardon, C., Zhang, C., Parker, L. E. (2015, May 1)*Adaptive human-centered representation for activity recognition of multiple individuals from 3D point cloud sequences. IEEE Xplore.* https://doi.org/10.1109/ICRA.2015.7139459

[3] *Syed, Z., Gilani, A., Mian. (n.d.). Learning from Millions of 3D Scans for Large-scale 3D Face Recognition.* https://openaccess.thecvf.com/content$_c vpr_2$018/ papers/Gilani$_L earning_F rom_M illions_C VPR_2$018$_p aper.pdf$

[4] Singh, A. D., Sandha, S. S., Garcia, L., Srivastava, M. (2019)*RadHAR. Proceedings of the 3rd ACM Workshop on Millimeter-Wave Networks and Sensing Systems - MmNets'19.* https://doi.org/10.1145/3349624.3356768

[5] Zhao, P., Lu, C. X., Wang, J., Chen, C., Wang, W., Trigoni, N., Markham, A. (2019, May 1)*mID: Tracking and Identifying People with Millimeter Wave Radar. IEEE Xplore.* https://doi.org/10.1109/DCOSS.2019.00028

[6] anakaraj, P., Jakkala, K., Bhuyan, A., Sun, Z., Wang, P., Lee, M. (2019, September 1).Simultaneous Tracking and Recognition through Millimeter Waves and Deep Learning. IEEE Xplore. https://doi.org/10.23919/WMNC.2019.8881354

[7] Atik, M. E., Duran, Z. (2021). *Deep Learning-Based 3D Face Recognition Using Derived Features from Point Cloud. Innovations in Smart Cities Applications Volume 4, 797–808.* https://doi.org/10.1007/978-3-030-66840-2$_6$0

[8] Zhao, P., Lu, C. X., Wang, J., Chen, C., Wang, W., Trigoni, N., Markham, A. (2021). *Human tracking and identification through a millimeter wave radar. Ad Hoc Networks, 116, 102475.* https://doi.org/10.1016/j.adhoc.2021.102475

[9] Palipana, S., Salami, D., Leiva, L. A., Sigg, S. (2021). *Pantomime. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 5(1), 1–27.* https://doi.org/10.1145/3448110

[10] Tiwari, G., Gupta, S. (2021). *An mmWave Radar Based Real-Time Contactless Fitness Tracker Using Deep CNNs. IEEE Sensors Journal, 1–1.* https://doi.org/10.1109/jsen.2021.3077511

[11] Wang, Y., Liu, H., Cui, K., Zhou, A., Li, W., Ma, H. (2021, June 1). *m-Activity: Accurate and Real-Time Human Activity Recognition Via Millimeter Wave Radar. IEEE Xplore.* https://doi.org/10.1109/ICASSP39728.2021.9414686

[12] Yu, C., Xu, Z., Yan, K., Chien, Y.-R., Fang, S.-H., Wu, H.-C. (2022). *Noninvasive Human Activity Recognition Using Millimeter-Wave Radar. IEEE Systems Journal, 1–12.* https://doi.org/10.1109/JSYST.2022.3140546

[13] Qi, C., Su, H., Mo, K., Guibas, L. (n.d.). *PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation.* https://openaccess.thecvf.com/content$_c vpr$ 2017/papers/Qi$_P ointNet_D eep_L earning_C VPR_2$017$_p aper.pdf$