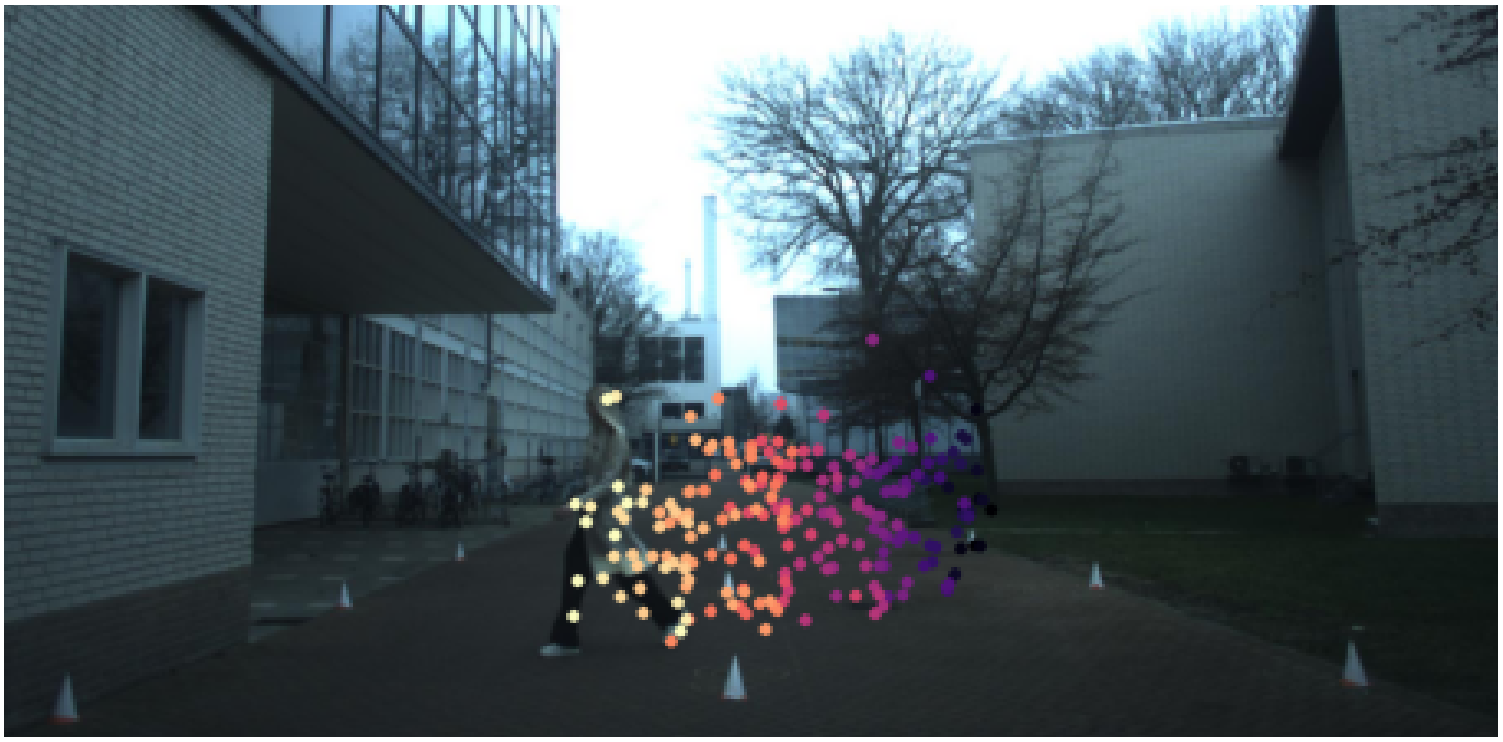
M A S T E R T H E S I S R E P O R T



Human Identification Using Automotive Radar

E M M A V A N S C H O T H O R S T - A L L E M E K I N D E R S

TU Delft

Human Identification Using Automotive Radar

Thesis report

by

E.A. van Schothorst - Allemekinders

to obtain the degree of Master of Science in Robotics
at the Delft University of Technology,
to be defended publicly on Thursday September 28, 2023.

Student number:	5608198	
Project duration:	December 1, 2022 – September 28, 2023	
Thesis committee:	Dr. H. Caesar,	TU Delft, supervisor
	Dr. A. Pálffy,	TU Delft, daily supervisor
	Dr. M. Mazo Espinosa	TU Delft,
	Dr. F. Fioranelli	TU Delft

TU Delft

Human Identification Using an Automotive Radar

E.A. van Schothorst - Allemekinders

***Abstract*—In this study, we perform human identification using accumulated radar point clouds in an outdoor scene. We employ PointNet as classification network and explore the impact of adding radars' non-spatial features as input, namely doppler velocity and radar cross section (RCS). Furthermore, we encode time as an additional time identity dimension to each point within the accumulated point cloud. We examine the effects of normalizing the RCS values, canonicalizing the spatial dimensions of the point cloud, as well as normalizing the doppler velocity with respect to this canonicalization. We examine three different PointNet configurations to understand the impact of the TransformNet blocks (T-Net) within the PointNet architecture on our six-dimensional radar data input. We have created a realistic outdoor dataset for training and evaluation purposes. Our approach of using the unnormalized six-dimensional radar data on the PointNet architecture without the two T-Net blocks achieves the highest performance of 73.4 $\%$ on our test set.**

***Index Terms*—Radar, Point Clouds, Deep learning, Human Identification, PointNet, Canonicalization.**

I. Introduction

Human Identification is the process of recognizing an individual based on their unique characteristics and features that set them apart from a larger group [1]. Human identification has many applications in various domains, such as security, surveillance, and pedestrian tracking. The ability of cameras to capture detail, representing a person through detailed pixel colors, makes it a logical choice to use for human identification [2]. However, this capability poses a significant threat to privacy. Extracting features from images, especially without explicit consent, can lead to potential misuse of personal information [3]. In the midst of this controversy, radar emerges as a potential alternative. Several properties of radar enable it to represent the environment efficiently and in a privacy-preserving manner. Unlike cameras, radar systems emit radio waves and process their reflections, a method that inherently omits capturing visual details of individuals. Since radar systems rely on radio waves, which have a longer wavelength than visible light, they can operate effectively in different lighting conditions, as well as in many conditions that can challenge other sensing methods, such as rain, fog, or snow. Moreover, radar sensors provide simultaneous measurements of position and radial velocity. This allows for the extraction of both spatial and dynamic features.

While radar data offers its set of advantages, it also presents challenges. Radar point clouds are relatively sparse compared to, for example, lidar sensors, making it harder to perform human identification. Several research efforts have tackled human identification with radar data [4], [5], [6], [7], [8]. However, these studies have been carried out over short distances and focused mostly on indoor environments. To overcome the sparsity problem with radar data, they prioritized not only extracting shape features but also walk characteristics - gait features- that can be used as human identifiers [9], [10]. Not only do radar sensors capture data in 3D spatial dimensions, providing depth and positional information that isn't always available in traditional 2D imaging, radar sensors are also able to capture the doppler velocities and radar cross section (RCS) values. These characteristics of radar facilitate the learning of gait characteristics [11]. Inspired by these works, we also tackle the radar human identification challenge. We accumulate radar point clouds to obtain a richer point cloud and add a time identifier to each point to capture gait features. Each point within our point cloud has six dimensions, mainly, the three spatial dimensions and additionally the radial velocity (often also called the doppler velocity), the Radar Cross Section (RCS) value, and the time identifier: $[X, Y, Z, D, RCS, T]$.

Another challenge with radar point clouds is one that is inherent to point cloud data in general. Point clouds are a collection of unordered points in space. Meaning that points within a point cloud do not inherently convey how they relate to one another. That is why most researchers transform the point cloud into a regular structure such as voxels or images prior to preprocessing them for neural network input. PointNet [12] overcomes this problem and is able to process point clouds directly. It sorts the input in a canonical order, by learning a transformation matrix using a mini-network called Transform-Net (T-Net). By using a symmetric function (max pooling) they make the network robust to point permutations. We adopt this network for the human identification task with radar data. However, since radar data does not only have spatial dimensions, the doppler and RCS dimensions might not benefit from this T-Net alignment [8], [6]. To fairly evaluate the impact of the T-Net on our radar data, we exclusively focused on the PointNet classifier architecture, investigating three distinct configurations to determine which setup yields the highest performance using our six-dimensional radar data.

Human identification also faces the so-called viewpoint problem [13]. This refers to the challenge of consistently recognizing an individual when the angle or perspective from which the radar signal interacts with them changes. Therefore, we investigated the effect of three normalization strategies to improve the model's performance. We canonicalized the point clouds and normalized the magnitude of the doppler velocity with respect to this canonicalization. We also normalized the RCS values.

An additional complication when attempting human identification with radar is the limited availability of public datasets.

Radar's ability to operate within various lighting and weather conditions presents a great opportunity to use radar in outdoor environments. However, since no datasets are publicly available with radar point clouds of humans walking in an outdoor scene, we created our own dataset. The dataset consists of camera, radar and lidar data from 52 volunteers walking in an outdoor environment.

Our main contributions are as follows:

1) **Architecture Exploration:** We explore three different variations of the PointNet architecture for our six-dimensional radar data and compare them.
2) **Data Enhancement:** We assess the impact of accumulating the point clouds and the introduction of a time identifier to each point within the accumulated point cloud.
3) **Normalization Exploration:** We experiment with three distinct normalization methods for the spatial dimensions, the doppler velocity, and the RCS values and evaluate their impact on the models' performance.
4) **Dataset Creation:** We introduce the first outdoor multimodal human identification dataset, including data not only from 4D automotive radar, but also from a Lidar, and camera sensor, with more than 50 participants. This dataset will help us in advancing our academic investigations.

II. Related Work

There are several methods using radar data for human identification. We categorize them into two categories, low-level radar approaches and point cloud-based approaches. Whereas the low-level radar approaches consist of all approaches that do not make use of a point cloud representation.

A. Low Level Radar Approach

RF-Capture proposed by [14], is able to capture the contour of a human body using radar data. They use a coarse-to-fine algorithm to efficiently generate 3D snapshots of RF reflections. They also faced the problem that a single snapshot of radar data fails to capture the complete bodily details. Therefore they detect body parts over multiple executive frames in which the person is moving towards the radar sensor. These retrieved body parts are then stitched together to obtain the entire human figure. This is used as input to a support vector machine (SVM) to perform the human identification task.

[4] propose MU-ID, an identification system that has the capability of recognizing multiple people based on lower limb motion features that they retrieve from low-level radar data. They create from the radar signals two-dimensional step segments that contain three main gait characteristics, mainly, step length, the distance between two lower limbs, and the instantaneous lower limb velocity. These 2D images are the input to a CNN classifier which performs the identification task.

B. Point Cloud Based Approaches

Many methods however convert radar data into point cloud representations. A radar point cloud is an unordered set of points. Each point contains information about the location in 3D space and often other information like doppler velocity and RCS.

To convert the point clouds from their irregular structure into a regular structure, [5] first voxelizes the points of potential human point clouds, to create an occupancy grid. By flattening the 3D data and converting each frame into a feature vector, they can feed it into a bi-directional LSTM (Bi-LSTM) network followed by a dense layer to perform the classification task. To ensure that the captured data is not too noisy or sparse, they limit their range to only a 5-meter distance from the radar. They are, to the best of our knowledge, the first to perform a human identification task by utilizing point clouds from an mmWave radar. Given that radar data is already sparse, voxelizing the point cloud can result in a majority of empty voxels. This extreme sparsity can be computationally inefficient and might not offer a meaningful representation advantage. Additionally, when selecting a specific grid resolution, it might be too coarse, which could lead to a loss in crucial details; or too fine, and the sparsity and computational inefficiencies become even more pronounced.

[6] resolve the point clouds irregular structure problem differently. They utilize five attribute networks that each take one of the five attributes: $[X, Y, Z, D, RCS]$ as input. This modular approach allows them to optimize each network for a specific attribute. After that, a feature fusion network is employed to fuse the features extracted from these attribute networks, resulting in a comprehensive representation suitable for the human identification task. However, their study is limited to an indoor scene with humans at a small distance of up to a maximum of 6 meters from the radar device. Another shortcoming is that they collect radar data with two devices from two different viewpoints at the same time, which is not a convenient approach if human identification should be performed outside, for example, in a moving car.

[7] created an extra input dimension named point flow. For each adjacent frame, they first match the points to the points in the previous frame. By subtracting the doppler velocities of these matched points, they create a new dimension describing the point flow. Their model is called HDNet, which uses a Graph Neural Network (GNN) backbone to extract features. The features are then fed to a Transformer block, to further aggregate the temporal features which are then used for classification. This inspired us to also create an extra dimension. As PointNet is not a sequence learner, we instead accumulate the point clouds. Instead of the complex retrieval of point flow for each point within the accumulated point cloud, we add a time identifier as an extra dimension to each point.

[8] created SRPNet, which consists of 4 modules. The first module is a modified version of PointNet. After the first T-Net block in the PointNet architecture, the doppler dimension is concatenated. Then they feed the global features

into an attention module which is followed by a Bi-LSTM. The classification is then performed using several MLP layers. They clarify that they add the doppler dimension after the first T-Net in PointNet as the doppler velocity does not satisfy the affine invariance. The doppler velocity can reflect the dynamic features of the human and should not be lost by the transformation from the first transform block. Meanwhile, [6] used PointNet combined with LSTM to evaluate their dataset. Instead of concatenating the doppler dimension after the first T-Net, they removed the two T-Net blocks in their PointNet-LSTM architecture and reported higher performance compared to using PointNet-LSTM with T-Net. We therefore adopted both approaches and focused solely on the PointNet classifier to make a fair comparison to investigate how T-Net affects our additional non-spatial features. Additionally, we also investigated removing only the first T-Net block. Our results confirm that utilizing PointNet without both T-Net blocks results in the highest performance for our six-dimensional radar data.

III. Dataset Generation

The adaptability of radar, especially its capability to perform consistently across different lighting and weather conditions, makes it a promising tool for outdoor scenarios. However, due to the lack of publicly available datasets that capture radar point clouds of people walking outdoors, we created our own dataset. The dataset aims to provide a comprehensive view of humans walking within an 18-meter radius in outdoor settings. Three sensors were utilized: camera, lidar, and radar. 52 volunteers were instructed to walk along 22 predefined paths, including trajectories where participants walked straight toward the sensors, moved perpendicular to the sensors' line of sight, and moved diagonally in relation to the sensors' position (see Figure 1).

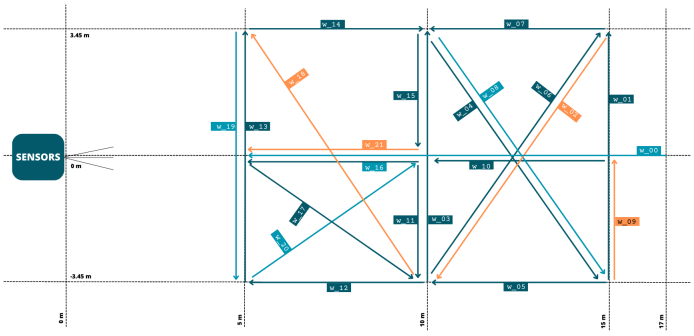


Fig. 1. The predefined walking paths in our dataset. Green = Train set, Orange = Val set, Light Blue = Test set.

This dataset was recorded with multiple use-cases in mind, leaving the possibility for performing identification also with other sensors or fusions. Similar to most identification-related related use-cases, i.e., security, or tracking of incoming pedestrians, we prioritized incoming trajectories. Between each defined path, the volunteer stopped walking. After the data recording, we manually annotated the data, removing the frames where the person was in standstill and adding the path identity to each frame. Metadata capturing significant appearance variations—such as individuals wearing hats or walking with hands in pockets—was included as metadata. For each person, around 104.2 seconds of walking data was recorded, which corresponds to around 1042 frames per person. The group of volunteers consisted of 41 males and 11 females. The data was recorded in February at day-time, however during twilight, the car headlights were put on to make sure the images were not too dark.

IV. Data Preprocessing

This section first describes the data preprocessing steps of how the human radar point clouds have been retrieved. Afterward, the three normalization techniques are detailed.

A. Radar Scans Accumulation

The first step of the data preprocessing involved finding the radar frame with the timestamp that is closest to each given image frame, thereby forming a precise pairing of data that correspond to the same temporal moments. The bisect algorithm was used which efficiently found the closest matching radar frames for each image frame based on their respective timestamps. The next stage of the data preprocessing involved accumulating additional radar scans for each matched radar frame. Specifically, for each radar frame that was aligned with an image frame, we accumulated up to 28 preceding radar scans (see Figure 2), which corresponds to approximately 2 seconds in real-time duration. In the resulting accumulated point cloud, each point was tagged with a time ID that ranges from 0, representing the current frame, to -28, which corresponds to the frame 28 scans earlier.

B. Human Point Cloud Extraction

Once accumulated, these point clouds first underwent filtering to include only the points within the walking area. The second step involved extracting the human radar points from the remaining points. As radar is sparse, we leveraged the dense point clouds from lidar to obtain a consistent representation of the centers of the human point clouds. The thesis candidate working on the separate lidar research obtained the lidar clusters by using DBSCAN. We utilized these lidar clusters to compute the centers in the lidar coordinate system and subsequently project them onto the radar coordinate system.

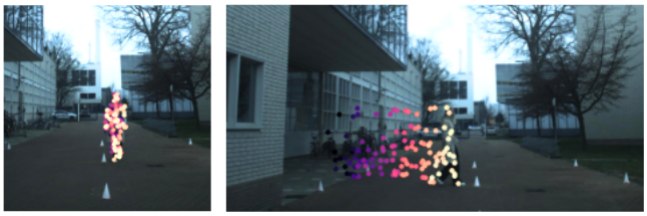


Fig. 2. Examples of the obtained accumulated human radar point clouds, where each color represent the time identifier ranging from 0 until -28.

All points within a close distance to the radar centers were then filtered to obtain the human radar point clouds. Note that $walk_{00}$ (see Figure 1) has been clipped up until 17 meters, as the lidar clusters did not stay consistent after 17 meters. Also, the first 10 frames and the last 5 frames of each walk have been removed to compensate for the acceleration from a standstill and the deceleration to a standstill for each individual. As our models require a fixed number of input points, we took the mean number of points for each accumulation as input. For 28 accumulations the mean number of points was 157. If the number of points within the point cloud was higher than the desired number, the extra points were deleted at random. If the number of points within the point cloud was lower than the desired number, additional points were randomly copied from existing ones to meet the count.

C. Normalization

Normalizing techniques can in some cases help machine learning models to converge faster and potentially reach better performance. Several normalization strategies were employed. An overview of these strategies will be given.

1) Spatial Orientation Normalization

The goal of the model is to associate the same features with the same person, regardless of their orientation. To make it easier for the model, we try to bring the spatial coordinates in a canonical space. To achieve this, the orientation of each point cloud has been aligned with the negative x-axis (see Figure 3). The first step included centering each point cloud around the origin by subtracting the [x,y]-value of the first point cloud center from each point within the entire point cloud. Subsequently, from a window of 5 consecutive frames, the centers were used to obtain the heading direction $\vec{h}$ of the point cloud by using Principal Component Analysis (PCA). Then, the angle between the heading direction of the point cloud and the negative x-axis was calculated using equation 1.

$$\theta = \arccos\left(\frac{\vec{h}\cdot\vec{r}}{\|\vec{h}\|\|\vec{r}\|}\right) \tag{1}$$

where $\vec{h}$ is the heading vector and $\vec{r}$ is the reference axis $[-1, 0]$.

This angle, θ, was then used to rotate the point cloud around the z-axis so that its heading aligned with the negative x-axis, as denoted in equation 2.

$$\vec{p}_{\text{normalized}} = (\vec{p} - \vec{c})R_z(\theta)^\top \tag{2}$$

Here, $\vec{p}$ is a point in the original point cloud, $\vec{c}$ is the center point, and $R_z(\theta)^\top$ is the rotation matrix.

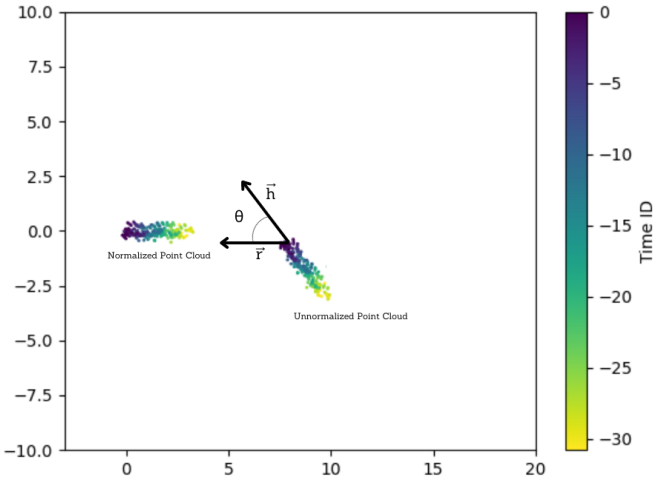


Fig. 3. Illustration of the point cloud spatial normalization, the colors indicate the encoded time, ranging from 0 until -28.

2) Relative Velocity Normalization

A person walking at the same speed but in different directions (relative to the radar sensor) will produce different Doppler shifts. This variability in the doppler velocity can complicate the task of recognizing patterns in the data, as the same object could appear different to the radar based on its direction of movement. Therefore we normalize the Doppler velocity to represent the doppler velocity as if a person is always walking toward the radar (so also in line with the normalization of the spatial dimensions). By normalizing Doppler velocities, the model gets a uniform representation of human motion, reducing variability in the input space (see Figure 4). To achieve this, the normalized relative velocity was calculated by dividing the magnitude of the velocity vector by the cosine of the angle between the velocity vector and the heading direction of the point cloud.

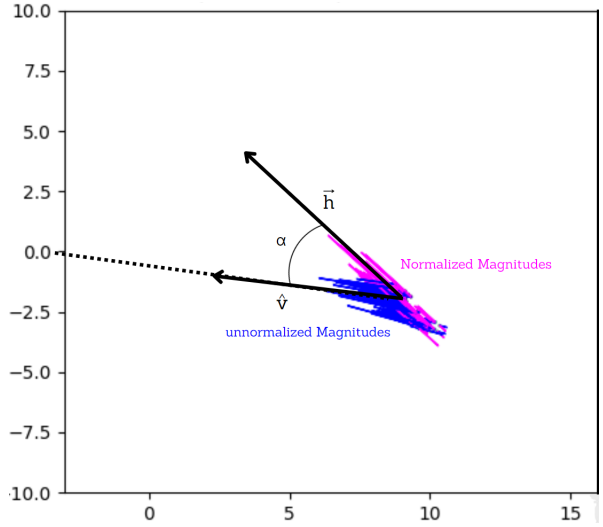


Fig. 4. Illustration of the relative velocity magnitude normalization.

First, the measured doppler velocity magnitude has been calculated. This magnitude represents how fast the object is moving, irrespective of its direction. The magnitude of the velocity vector **v**, for each point in the point cloud has been calculated as follows:

$$D = \|\mathbf{v}\| \tag{3}$$

Note that the symbol D represents the unnormalized doppler velocity magnitude. The direction vector $\hat{v}$ is obtained by dividing the velocity vector **v** by its magnitude D. This gives a unit vector that points in the same direction as **v** but has a magnitude of 1. It captures the direction of movement without considering the speed. The direction vector $\hat{v}$ has been obtained as follows:

$$\hat{v} = \frac{\mathbf{v}}{D} \tag{4}$$

The cosine of the angle between the direction vector $\hat{v}$ and the heading vector $\vec{h}$ is given by equation 5.

$$\cos(\alpha) = \hat{v} \cdot \vec{h} \tag{5}$$

Finally, the doppler velocity D has been adjusted based on its direction of movement. The normalized doppler velocity D_{norm} could then be calculated using the equation:

$$D_{norm} = \frac{D}{\cos(\alpha)} \tag{6}$$

When the angle between the direction of motion of a point cloud and the line of sight from the sensor is close to 90 degrees, and the measured relative velocity is relatively high, the normalization will result in extreme values. These extreme normalized values are clipped to the most commonly observed maximum value during a walk toward the sensor.

3) RCS Z-score Normalization

The Radar Cross-Section (RCS) values were normalized by using Z-score normalization, see equation 7. The effect of this normalization is that the RCS values will have a mean of 0 and a standard deviation of 1.

$$rcs_{norm} = \frac{(rcs - rcs_{mean}))}{rcs_{std}} \tag{7}$$

V. MODELS

PointNet is a deep learning architecture developed by [12] that made it possible to directly process a point cloud without requiring a grid structure. The first part of the network consists of a mini-network called TransformNet (T-Net). T-Net is a mini-network that predicts an affine transformation matrix to align the point set into a canonical space. For instance, consider the human radar point clouds. As the human walks, the radar generates point clouds that capture the person in various spatial orientations. T-Net aims to learn a transformation matrix to remove any arbitrary rotation or translation by transforming the point clouds into a canonical - logical - orientation. Thus helping the network focus on the structure of the data rather than the orientation or position in space. Although this example is an illustration and is limited to the spatial 3D space, note that T-Net can also learn higher dimensional transformation matrixes, so it can also have input of higher dimensions. After the matrix multiplication, the network uses a shared multi-layer perceptron (MLP) which learns point-wise features. This is followed by the feature alignment network, the second time T-Net is used within the PointNet architecture. It predicts a feature transformation matrix to align the different point cloud features.

This is then followed by several MLP layers to further refine the features. Then a max-pooling layer is utilized to extract

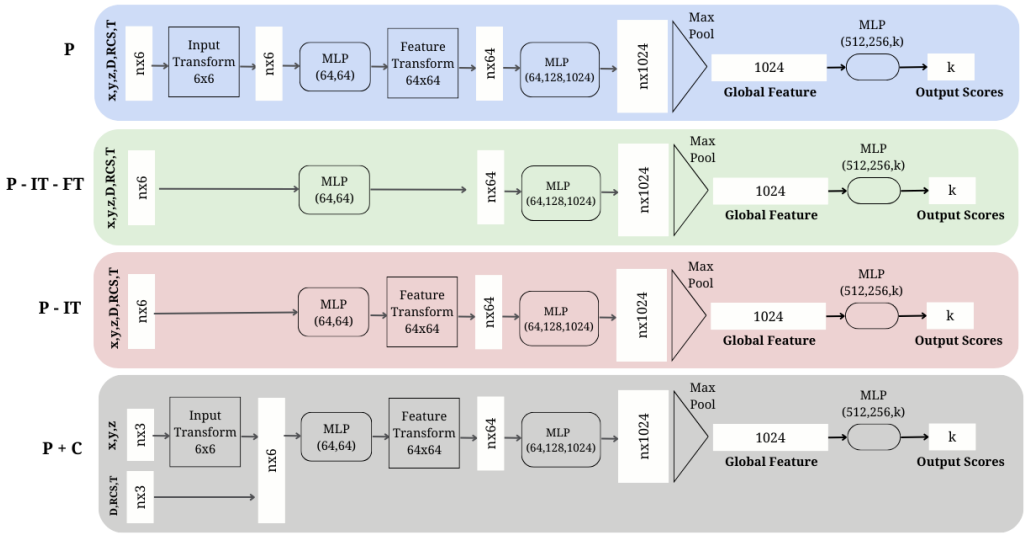


Fig. 5. OVERVIEW OF COMPARED METHODS - PointNet (**P**), PointNet without the input transform (**P-IT**), PointNet without the input transform and without the feature transform (**P-IT-FT**). PointNet in which the non-spatial dimensions are concatenated after the first input transform (**P+C**).

the global features of the point cloud. In the final step, a fully connected neural network is used with three layers to map the global feature vector to k classification scores. All layers include ReLU and batch normalization, with the last layer as the exception.

For our 6D radar point clouds $[X, Y, Z, D, RCS, T]$, the doppler velocity and radar cross section values do not satisfy the affine invariance [8]. When the points are rotated and translated, the measured doppler velocity and RCS values would have changed for the same object. To address this problem, three different configurations are investigated. In the first configuration (**P - IT**), the input transform block is removed from the PointNet (**P**) architecture. In the second configuration (**P-IT-FT**), both the input transform block as well as the feature transform block are removed. In the third configuration (**P+C**), the model takes as input the 3D spatial dimensions of the point cloud $[x, y, z]$. The non-spatial dimensions are concatenated after the input transform matrix multiplication with the points.

VI. Experiments and Evaluation

We have split the data into a train, validation, and test set. For the validation and test sets, 4 walks each were used, see Figure 1. The train set contains the remaining 14 walks. We trained the models for 25 epochs. The widely used classification loss cross-entropy loss is used. If the model uses T-Net, a regularization loss is added to the classification loss to make sure that the learned transformation matrixes are close to orthogonal.

$$L_{\text{reg}} = \alpha \left\| I - AA^T \right\|_F^2 \tag{8}$$

where I is an identity matrix and α is a weight (set to 0.0001) indicating the impact the regularization loss should have.

Consequently, the total loss is given by:

$$L_{\text{total}} = L_{\text{CE}} + L_{\text{reg}} \tag{9}$$

where L_{CE} is the cross entropy loss.

As an optimizer, the ADAM optimizer is used with an initial learning rate of 0.001, and a momentum of 0.9. A batch size of 32 has been used. Every 10 epochs, the learning rate is divided by 2. On the last layer, a dropout layer is used with a keep ratio of 0.7. A decay rate for batch normalization is used which starts with 0.5 and gradually increases to 0.99. For all experiments, except the last experiment, we performed the classification task for 5 persons. In all experiments except for the experiment in which we vary the number of accumulated point clouds, there are 28 accumulated point clouds used. Since our method requires a fixed number of input points, we used the mean number of points within 28 point clouds, which corresponds to 156 points in total. Since in the final epochs, the model starts to oscillate around an equilibrium, we use the weighted moving average smoothing function in Tensorboard to have comparable validation scores. A modest smoothing factor of 0.3 was used to obtain the most balanced results. As an evaluation metric, the F1 score has been chosen. As some people might walk faster than others, resulting in more frames, we average the F1 score in the following way. The F1 score is calculated for each class (i.e. person) independently but when it averages them, it uses a weight for each class's score that depends on the number of true instances for each class. This takes into account the imbalance of the classes.

A. Method Investigation

To investigate which model delivers the highest performance for our six-dimensional input $[x, y, z, D, RCS, T]$, we input this six-dimensional data to the models as explained in Section V. The results can be seen in Table I.

TABLE I
Overall F1 Score - Validation Set

	F1 Score
	Unnormalized Input
P - IT - FT	**72.7**
P - IT	62.3
P + C	58.0
P	53.4

On the validation set, our method without both transform blocks (**P - IT -FT**) achieves the highest performance of 72.7%. On the test set, which can be seen in Table II, the method without both T-Net blocks (**P - IT -FT**) also achieves the highest F1 score of 73.4%. On the test set, both **P - IT** and **P - IT - FT** achieve the highest performance for person 4 and the lowest performance for person 1. Using solely PointNet results in the lowest performance for our six-dimensional input.

TABLE II
Individual Accuracy and Overall F1 Score - Unnormalized Inputs - Test Set

	P1*	P2*	P3*	P4*	P5*	Overal F1 Score
P - IT - FT	**66**	**74**	**69**	**84**	74	**73.4**
P - IT	56	58	59	80	62	63.0
P + C	66	50	44	83	51	59.5
P	46	56	50	41	**87**	55.2

* Represents accuracy per individual.

Effect of the Normalized six-dimensional input

To demonstrate the impact of the three normalization strategies that were described in Section IV, we perform experiments with the normalized data. We first feed the models the normalized input [XYZ_{norm},D_{norm},RCS_{norm},T] as described in Section V. The results can be seen in Table III and Table IV. On the validation set, the method **P-IT** achieves the highest performance, while on the test set the method **P-IT-FT** achieves the highest method.

TABLE III
OVERALL F1 SCORE - VALIDATION SET

	F1 Score
	Normalized Input
P - IT - FT	51.1
P - IT	57.3
P + C	48.8
P	51.7

However, all methods decline in performance when all three normalization techniques are used compared to the unnormalized input, for both the validation and test set. So it can be concluded that using these three normalization techniques together is ineffective for all methods.

TABLE IV
INDIVIDUAL ACCURACY AND OVERALL F1 SCORE - NORMALIZED INPUTS - TEST SET

	P1*	P2*	P3*	P4*	P5*	Overall F1 Score
P - IT - FT	**49**	51	**72**	**56**	35	**51.5**
P - IT	25	44	71	34	23	40.2
P + C	46	**69**	56	25	**65**	48.9
P	28	37	71	47	39	41.9

* Represents accuracy per individual.

B. Ablation Studies for P-IT-FT

From these experiments so far can be concluded that **P-IT-FT** is the most optimal for our six-dimensional input data. Further experiments mentioned here will be ablations for this method. Results from more experiments with the other methods can be found in Appendix A.

1) Effect of Individual Normalization

To further investigate the effect of normalization on our chosen method (**P-IT-FT**), we explored the effect of the normalization methods individually.

TABLE V
EFFECT OF NORMALIZATION - VALIDATION SET

		F1 Score
Inputs	XYZ	47.9
	XYZ_{norm}	36.1
	XYZ + D_{norm} + RCS + T	69.7
	XYZ + D + RCS_{norm} + T	70.3
	XYZ_{norm} + D_{norm} + RCS_{norm} + T	51.1
	XYZ + D + RCS + T	**72.7**

As can be seen in Table V, the XYZ normalization degrades the performance from 47.9% to 36.1%. From this can be concluded that the XYZ normalization was ineffective. However, when only Doppler or RCS is normalized individually or together, the model degrades slightly in performance. Using the unnormalized data is the most optimal for our method.

2) Effect of Non-Spatial Dimensions

To investigate the effect of each non-spatial dimension individually and in relation to one another, we fed our highest performing model (**P - IT - FT**) a non-spatial dimension, next to its spatial dimensions.

TABLE VI
F1 SCORE - UNNORMALIZED INPUTS- VALIDATION SET

	F1 Score
XYZ	47.9
XYZ + RCS	62.7
XYZ + D	62.2
XYZ + T	53.4
XYZ + D + RCS	57.4
XYZ + D + T	67.7
XYZ + RCS + T	68.2
XYZ + D + RCS + T	**72.7**

The results can be seen in Table VI. Adding solely the time identifier achieves a higher performance compared to only using XYZ (47.9 % vs 53.4%). Adding solely doppler or RCS increases the performance around the same number, respectively 62.2 % and 62.7%. Adding the time identifier in combination with either doppler or RCS also increases the performance further. These results demonstrate that the model learns gait features from the time identifier.

3) Effect of Accumulation

To investigate the effect of the number of accumulations we tried the following number of accumulated scans: $[0, 1, 2, 3, 4, 5, 6, 8, 10, 12, 14, 16, 20, 24, 28]$. As our model required a fixed number of input points (also mentioned in Section IV), we took the mean number of points for each accumulation as input. We stopped at 28 accumulations as this corresponded to two seconds in real time and the performance only slightly increased from here. In our method, we used the 28 accumulations since memory storage was not an issue. However, if efficiency is a priority, after 16 accumulations, the performance uptick is minimal.

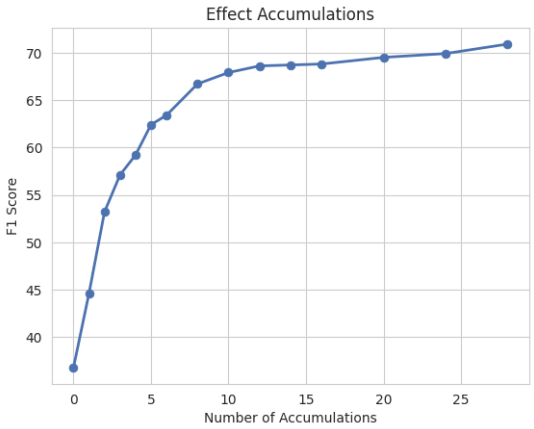


Fig. 6. Plot that illustrates the effect of accumulations.

4) Effect Number of Classes

To demonstrate the impact of the number of classes on our model, we performed the classification task for the following number of persons: $[3, 5, 10, 20, 30, 40, 52]$. As can be seen in Figure 7, as the number of persons increases, the identification task becomes more difficult. Nevertheless, as can be seen in the figure, the model stays predicting better compared to random guessing.

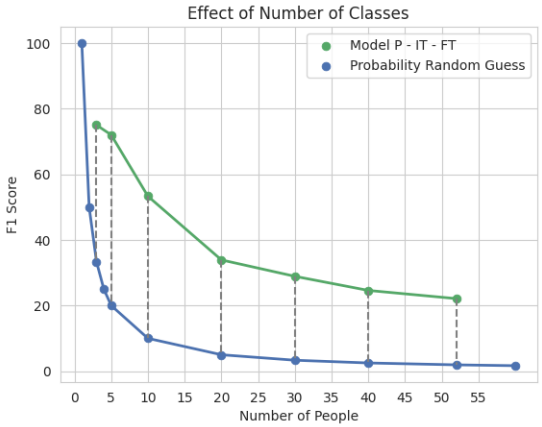


Fig. 7. Plot that displays the effect of the number of classes to be predicted. Our method consistently outperforms random guessing across the experiment.

VII. Discussion And Limitations

The findings in Table I and Table II demonstrate that for our six-dimensional input, the PointNet architecture without both T-Nets (**P-IT-FT**) on our unnormalized inputs achieves the highest performance. The T-Net is designed to perform rotation and translation transformations on input data. However, the properties of the doppler and RCS dimensions might not be well-suited for such transformations. As both dimensions do not satisfy the affine invariance, the T-Net blocks might dilute or misrepresent them, therefore leading to higher performance when the blocks are not present. This also explains why the other two modified networks achieve higher performance compared to using the original PointNet model with the six-dimensional data. However, an interesting observation is that PointNet achieves the highest performance for person 5. This person was walking with their hands in their pockets, potentially making their body silhouette more distinguishable. The T-Net blocks in PointNet could have played a crucial role in detecting this specific spatial pattern.

For the highest performing method (**P-IT-FT**), it can be seen in Table V that the normalization of Doppler and RCS does not reduce the performance much, however, the issue lies with the normalization of the spatial dimensions. Maybe this happened because the rotation of the point cloud was done for the entire accumulated point cloud. The degrees of how much the point cloud should be rotated were based on the moving direction of 5 executive center points. The assumption was that the person walked overall in the same direction. Using 5 executive frames can already cause misalignment if there are sudden movements or slight variations. Also rotating the entire accumulated point cloud as a single entity, as opposed to rotating individual frames, could lead to inaccuracies. Each frame might have its unique orientation nuances, which would be averaged out or even ignored when rotating the whole cloud. This might cause misalignments. A potential issue with the doppler normalization might be that when there were extreme values in the doppler dimension due to an angle close to 90 degrees between the line of sight and the direction of motion, we clipped them. This might oversimplify the complex dynamics of the doppler dimension. The performance of the model was not improved by the RCS normalization strategy but did not suffer much as well. One potential reason the normalization did not enhance performance is that, while the normalization changes the scale, it preserves the original data distribution. If the model is more influenced by distribution shape than by scale, normalization's impact is minimal.

For the unnormalized coordinates, when only RCS, Doppler, or the time identifier were added together with the spatial dimensions, they all reached higher performance compared to only using the spatial dimensions. This demonstrates that these dimensions are effective for human identification. Also, it shows that the encoding of the time to each point helps the model learn gait features. Remarkably, adding Doppler together with RCS without a time identifier results in a lower performance compared to when they are fed to the model individually. However, when the time identifier is added as well, the highest performance is reached. A potential reason can be that the raw doppler and RCS together cause feature interference and make it harder for the model to find the underlying patterns, but when the time identifier is added, it finds the underlying patterns more easily.

We will now address the challenges associated with the walking scene (Figure 1). The walking scene is 10 meters wide and ranges up to 18 meters. It is only within these specified dimensions that the model is exclusively evaluated. If a person is further away from the radar sensor, the data becomes more sparse which might lead to lower performance of the model. Another limitation is the presence of clutter points. In the walking environment, there is a wall that leads to clutter points. Also, there were cones placed in the walking environment to indicate where the person should walk. These cones lead to some clutter points when the person is walking close to it.

Following the predefined path, the volunteers pause consistently as they reach a new path number. When people start to walk, there are fewer scans, and thus fewer points available to learn from. Therefore it was chosen to discard the first 10 frames and the last 5 frames of every walking path. However, with 28 accumulations, there are still existing frames with fewer points which might influence the performance. Another issue with the obtained human point clouds might be the

fixed number of input points. The random down-sampling of the points might lead to the removal of important key defining points. While the up-sampling simply copies existing points and might mislead the model into focusing on non-key features, leading to lower performance.

VIII. Conclusion

Our research demonstrates that radar data, often perceived as challenging, can be effectively used for human identification. By making strategic adjustments to the PointNet system and thoughtfully creating the appropriate input, we have optimized PointNet's capability to identify humans using radar data. In conclusion, this study demonstrated that a modified PointNet architecture without both T-Net blocks results in the highest performance for our six-dimensional input data. Our experiments demonstrate that our canonicalization of the spatial dimensions, the normalization of the doppler velocity with respect to this canonicalization, and the normalization of RCS were proven to be ineffective. In contrast, the accumulation of point clouds and the addition of a time identifier to each point has proven to be effective for our model. This means that using this approach gait features can be picked up by the model. Additionally, a dataset with camera, radar, and lidar data of people walking in an outdoor scene has been established which can enhance future internal research. This study emphasizes the value and potential of radar data when utilized efficiently. This paves the way for effectively using radar data in which PointNet is combined with other deep learning architectures to enhance the feature learning even more.

References

[1] N. Khamsemanan, C. Nattee, and N. Jianwattanapaisarn, "Human Identification from Freestyle Walks Using Posture-Based Gait Feature," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 1, pp. 119–128, 1 2018.

[2] Z. Ming, M. Zhu, X. Wang, J. Zhu, J. Cheng, C. Gao, Y. Yang, and X. Wei, "Deep learning-based person re-identification methods: A survey and outlook of recent works," 10 2021. [Online]. Available: http://arxiv.org/abs/2110.04764

[3] G. Gonzalez-Fuster, M. Nadolna Peeters, and European Parliament. Directorate-General for Parliamentary Research Services., "Person identification, human rights and ethical principles," Tech. Rep., 2021.

[4] X. Yang, J. Liu, Y. Chen, X. Guo, and Y. Xie, "MU-ID: Multi-user Identification Through Gaits Using Millimeter Wave Radios;," *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, pp. 2589–2598, 2020.

[5] P. Zhao, C. X. Lu, J. Wang, C. Chen, W. Wang, N. Trigoni, and A. Markham, "MID: Tracking and identifying people with millimeter wave radar," in *Proceedings - 15th Annual International Conference on Distributed Computing in Sensor Systems, DCOSS 2019*. Institute of Electrical and Electronics Engineers Inc., 5 2019, pp. 33–40.

[6] Z. Meng, S. Fu, J. Yan, H. Liang, A. Zhou, S. Zhu, H. Ma, J. Liu, and N. Yang, "Gait Recognition for Co-Existing Multiple People Using Millimeter Wave Sensing," *AAAI Conference on Artificial Intelligence*, 2020. [Online]. Available: https://github.com/mmGait/people-gait.

[7] Y. Huang, C. Gu, Y. Fu, C. Zhuo, Y. Wang, K. Shi, and Z. Shi, "HDNet: Hierarchical Dynamic Network for Gait Recognition using Millimeter-Wave Radar," 2022. [Online]. Available: https://www.researchgate.net/publication/364987772

[8] Y. Cheng and Y. Liu, "Person Reidentification Based on Automotive Radar Point Clouds," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, 2022.

[9] W. Sheng and X. Li, "Multi-task learning for gait-based identity recognition and emotion recognition using attention enhanced temporal graph convolutional network," *Pattern Recognition*, vol. 114, p. 107868, 6 2021.

[10] J. E. Cutting and L. T. Kozlowski, "Recognizing friends by their walk: Gait perception without familiarity cues," *Bulletin of the Psychonomic Society*, vol. 9, no. 5, pp. 353–356, 1977.

[11] A. K. Seifert, M. G. Amin, and A. M. Zoubir, "Toward Unobtrusive In-Home Gait Analysis Based on Radar Micro-Doppler Signatures," *IEEE transactions on bio-medical engineering*, vol. 66, no. 9, pp. 2629–2640, 9 2019.

[12] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation," *CoRR*, 12 2016. [Online]. Available: http://arxiv.org/abs/1612.00593

[13] P. Limcharoen, N. Khamsemanan, and C. Nattee, "Gait recognition and re-identification based on regional LSTM for 2-second walks," *IEEE Access*, vol. 9, pp. 112 057–112 068, 2021.

[14] F. Adib, C.-Y. Hsu, H. Mao, D. Katabi, and F. Durand, "Capturing the Human Figure Through a Wall," *ACM Transactions on Graphics*, vol. 34, pp. 1–13, 2015.

APPENDIX A
ADDITIONAL RESULTS

TABLE VII
F1 SCORE - UNNORMALIZED INPUTS - VALIDATION SET

		Methods			
		P - IT - FT	P - IT	P + C	P
Inputs	XYZ	47.9	49.4	51.9	51.9
	XYZ + D	62.2	58.5	45.6	57.7
	XYZ + RCS	62.7	53.9	51.8	49.1
	XYZ + T	53.4	52.6	44.8	50.1
	XYZ + D + RCS	57.4	56.7	**67.3**	55.4
	XYZ + D + T	67.7	59.3	48.7	**61.3**
	XYZ + RCS + T	68.2	61.1	66.1	57.8
	XYZ + RCS + D + T	**72.7**	**62.3**	58.0	53.4

Table VII demonstrates that not every method achieves the highest performance when all input dimensions are added. However, adding all six dimensions at once always results in a higher performance compared to only using XYZ. When only the spatial dimensions are fed to the models, the behavior of **P+C** and **P** are of course the same because, without additional non-spatial dimensions, they have the same architecture. Method **P-IT-FT** and **P-IT** achieve lower performance, but **P-IT-FT** significantly lower. This happens because the spatial dimensions benefit from the T-Net, which is lacking in the latter model. Additional dimensions always improve the performance for **P-IT-FT** and **P-IT** compared to only feeding X,Y,Z. Remarkably, PointNet **P** achieves lower results when only RCS or only the time identifier is given in addition to the spatial dimensions. Meanwhile, in combination, **P** is able to achieve a bit higher performance. Both **P-IT-FT** and **P-IT** achieve the highest performance when all dimensions are added.

TABLE VIII
F1 SCORE - NORMALIZED INPUTS - VALIDATION SET

		Methods			
		P-IT-FT	P-IT	P + C	P
Inputs	XYZ_{norm}	36.1	30.8	29.0	29.0
	$XYZ_{norm} + D_{norm}$	38.3	34.4	31.1	32.5
	$XYZ_{norm} + RCS_{norm}$	51.3	51.6	30.3	38.7
	$XYZ_{norm} + T$	45.7	43.9	34.5	32.5