UCL

Final Thesis of Human Body Reconstruction from multiple levels of millimeter-wave radar data representation

Author
RUIHONG LIU

Student Number
23071850

Supervisor
Prof.CHRIS XIAOXUAN LU

Second Supervisor
Prof. PAUL BRENNEN

September 1, 2024

Final Project Dissertation
Msc in Integrated Machine Learning System
Dept. of Electrical and Electronic Engineering

Abstract

This paper presents the SimpleCFA model, a neural network that estimates 3D human meshes under challenging conditions such as poor lighting and low visibility environments, including fog and smoke. Millimeter-wave (mm-wave) radar signals can penetrate clothing and reflect off the human body. Also, mm-Wave has the benefit of detecting the small movement of the human body under high-frequency conditions.This model introduces a method for predicting human meshes using only radar signals as input data. Furthermore, the model is designed to work with a single person.
Converting mm-wave radar signals into a 3D human mesh is a highly under-constrained problem. To address this challenge, the model employs:
1) three convolutional neural networks and a feature pyramid network for feature extraction.
2) a multi-head self-attention mechanism to process different channels of information. This allows the model to focus model in different directions.
3) Bayesian optimization to determine the optimal weights for the model. Finding the suitable weight of loss function to improve the performance of the model.
The result has successfully shown that radar raw data could be used to 3D reconstruct human mesh directly, and the results demonstrate that the error in vertex prediction is around a 10-centimeter range, while joint prediction error is around a 20-centimeter range. The model is also capable of recognizing different poses and genders.
All the codes are presented in https://github.com/Ruihong-Liu/Final-project$_r adar_n avigation.git$

Contents

1 Introduction and Problem Statement

Predicting and estimating a full 3D human mesh, capturing both the body shape and pose of the human body is still a challenging task in the computer vision field. Recent work is based on three basic devices, which are the camera, Lidar, and radar.

Most researchers are using the camera-based method to convert a 2D RGB image data into a 3D mesh[1]. The reason for that is because camera methods are normally cheaper on manufacturing cost and smaller on size[2] than the other two devices. Also, taking the benefit of science and technology improvement, small cameras now have high image resolution which could provide more detailed information. Therefore, most of the auto-vehicle and air-drone manufacturers rely on using cameras to reconstruct human 3D mesh. The latest work has shown the possibility of using a vision transformer to convert a 2D image into a 3D mesh using image data only [3]. This method has successfully reduced the error of the 3D mesh within the 2mm range. Although it has advantages in cost and size, it can not work in poor light conditions or extreme weather conditions. Also, any covering on the human body such as clothes could have a significant effect on the accuracy.

Other researchers decided to use LiDAR because LiDAR can provide the most accurate 3D measurement over all three devices. LiDAR basically receives the reflection signal of infrared light emission. As light emission and receiving are point-to-point processes. Therefore, LiDAR can provide the most accurate 3D point cloud data for detection tasks[4]. Recent works have successfully used point clouds to achieve the human body and pose estimation. The procedure could be working in real-time pose tracking within 50 frames per second[5]. Although LiDAR could perform the best on the accuracy of the 3D reconstruction, the manufacturing cost is high, and small particles such as sand, and rain, could have a significant effect on the reflection of infrared light, which reduces the performance of the LiDAR.

Camera and LiDAR have limited performance under extreme weather conditions and poor light conditions. Radar which uses a radio frequency(RF) sensing system has filled in the gap. Radar has demonstrated it through wall human detection and pose estimation[6]. This method has proved that RF signals could travel through walls, and clothes and reflect off the human body. The problem with using radar is low resolution. Compared with camera and liDAR, radar has a much lower spatial resolution. This leads to the open question of whether it is possible to use radar for 3D reconstruction.

This report aims to 3D reconstruct human body mesh using only mm-wave raw data as the input. To achieve that, this paper introduces SimpleCFA, a neural network framework to convert radar raw data into 3D meshes. The model proves that the RF signal (transformed into the heat map) can convert into a 3D mesh directly without turning into a point cloud. It also predicts the gender to show more gender features on 3D human body mesh. Further, although the model is missing hand information, it could be predicted by controlling the loss of pose. To clarify, this model is working on a single person with a single frame prediction, not including any time series problems.
Using Rf signal from radar only is a difficult job. Although the RF signal is bouncing off from the human body, the reflection signal could be away from the detector rather than closer to it[7]. Directly using RF signals may not be possible, because RF signals do not

directly contain information about space information. Therefore, the RF signal needs to be converted into raw Image data (Heatmap) first by using Fast Fourier transformation and other application programming interfaces.

To tackle all the challenges and problems above. In this paper, we first developed a module that collects different dimensional information from the radar raw image. Three layers of CNN layers are used because the down-scaling strategy could benefit from a gradual reduction in the spatial resolution of the feature maps while extracting higher-level, more abstract features. As the input data has 31 channels, the networks need to filter the key features and focus on capturing global patterns rather than local details. Then all filtered key feature maps become the input into an FPN layer. The purpose of that is to allocate these feature maps at different scales to form new feature maps with rich semantic information. FPN could merge high-resolution, low-level features with low-resolution, high-level features, creating a set of multi-scale feature maps. Furthermore, the FPN model is integrated by adding a multi-head self-attention mechanism to further enhance the network's ability.

Overall, the model is trained using supervised learning algorithms. Radar raw image is used as the input value to the model, and vision to provide the ground truth for supervision. Ground truth 3D mesh is collected using Vicon and human joint, pose parameters are collected using image and SMPL-X model.

The data are collected and processed in the University of Edinburgh lab with 20 participants and each participant is doing exactly the same 50 actions. To reduce the training time and save some computing resources, this project is only using 9 participants' data, 60 percent for training, 10 percent for validation, and 30 percent for testing. Overall, without camera calibration, the vertices error achieves 7mm, and the pose of the human body could be successfully predicted. Although elbow joint prediction is not stable, overall the pose estimation achieves the goal.

2 Related Works

Although this report focuses on using raw radar data directly, some of the point cloud methods could still be valuable references for developing the model. Those models give some idea about how the 3D mesh could be converted without predicting a huge number of vertices directly from raw radar data.

2.1 mmMesh: towards 3D real-time dynamic human mesh construction using millimeter-wave

This previous work focuses on using point clouds to generate a real-time 3D human mesh estimation system. The input of this model is all feature vectors, in other words, all the 3D points consist of X, Y, and Z values. The paper demonstrates that low-resolution radar may only give hundreds of points but to estimate a full 3D human mesh may require predicting thousands of vertices on a human mesh-based information. Therefore, they incorporate the Skinned Multi-Person Linear Model as an additional constraint[8]. SMPL allows the model to use a limited number of parameters consist different types of information to represent the whole 3D human mesh instead of estimating the location of thousands of vertices. Figure 1 is the overall model design of mesh paper.

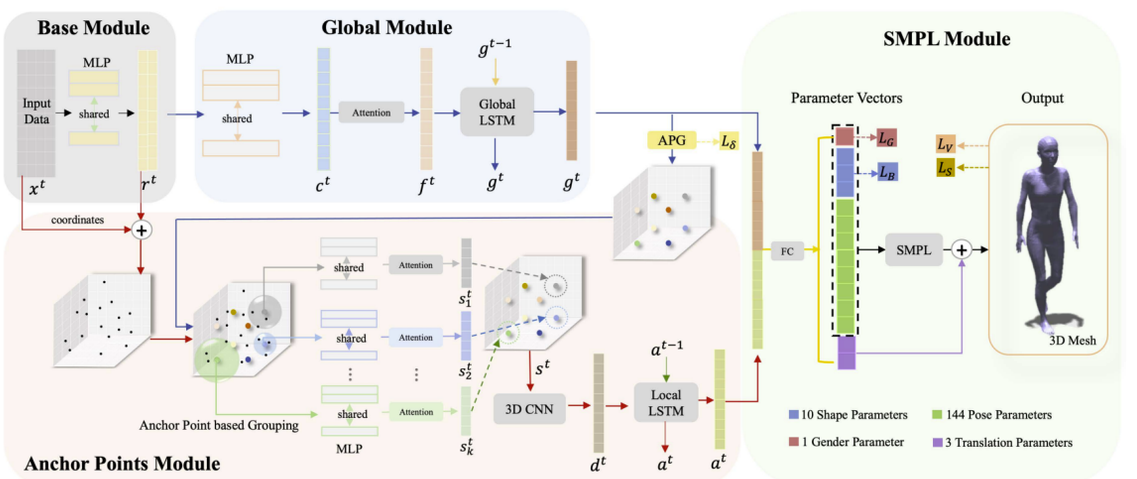

Figure 1: The flow chart of the mmMesh model design[8]

2.2 RF-Based 3D Skeletons

This is the first system that infers a 3D human skeleton from RF signals. The model is designed based on convolutional neural network architecture to perform. The paper demonstrates that CNN could be used to capture the 3D skeleton feature and key points such as head, shoulder joints, hand joints, and so on[9]. Figure 2 is the model design for this paper, the author uses RGB images to generate the ground truth of the human image skeleton, using Region Proposal Network to focus on single-person if multiple people in the environment. The accuracy of this model has an error within 4.2 centimeters on the X-axis and 4.0 centimeters on the Y-axis. This model is suggesting an idea about using 3D CNN to capture the key feature from the dataset.

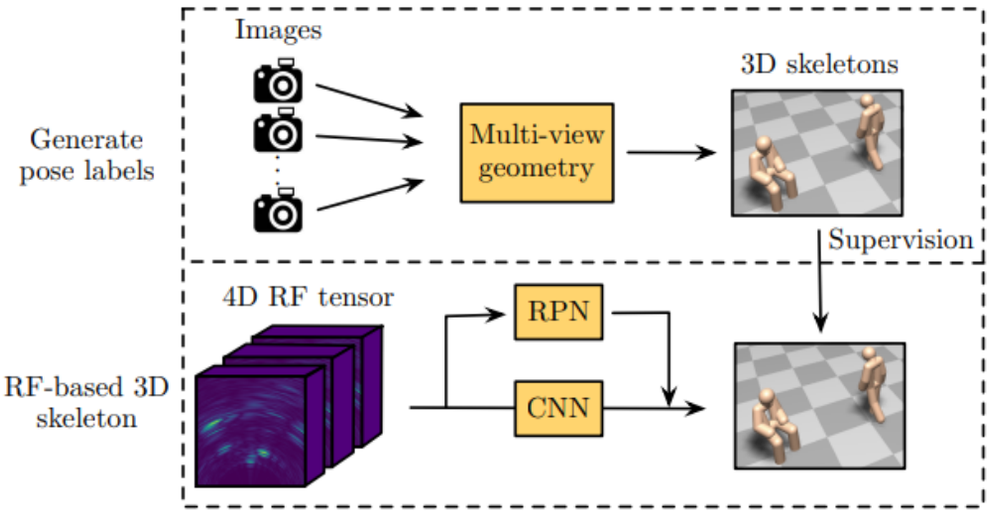

Figure 2: The flow chart of the RF-Based 3D Skeletons model design[9]

2.3 Convolutional Neural Network

The convolutional neural network is designed in 5 main parts, a Convolutional Layer, Activation Function, Polling layer, Fully connected Layer and Loss function, and Optimization. The convolutional layer is the core of the CNN where multiple kernels extract features from multiple channels. Then CNN applies a weighted sum over local regions of the input result in a feature map. The main purpose of CNN is to capture the local features of the input such as edges[10]. In this case, CNN is used for capturing all different local features from all different channels. Figure 3 is shown how CNN is capturing features through the window:

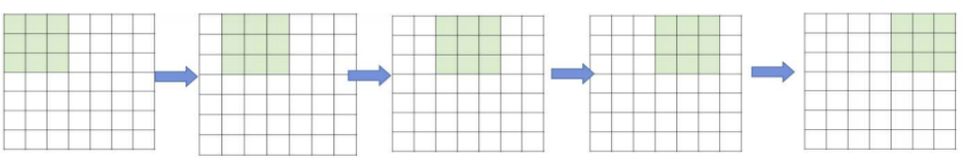

Figure 3: A diagram demonstrate how CNN is capturing the features [11]

2.4 Feature Pyramid Network

The main purpose of FPN is to build a pyramid structure that extracts features at different scales. FPN is typically integrated with convolutional neural networks (CNNs), such as ResNet, R-CNN, etc [12]. Because radar provides limited information on multiple channels, FPN can extract them well and combine high-level semantic features with low-level detailed features, providing rich representations across different scales.

2.5 Skinned Multi-Person Linear Model (SMPL)

SMPL factors the human mesh into a person-dependent shape vector and pose-dependent 3D joint angles[13]. SMPL requires four parameters, shape, pose, translation, and root orientation. shape requires 10 parameters, pose requires 63 parameters, translation needs

3 parameters and root orientation needs 3 parameters. In total, 79 parameters are required to input to the SMPL and this is the reason for output 79 outputs from the fully connected layer. For SMPL, if the vertex vector V and skeleton vector S can be obtained by feeding those four parameters into SMPL, the equation is shown below[14]:

$$[V^T; S^T] = SMPL(P^T, B^T; G^T) + T^T)$$

Where P is the pose vector, B is the shape vector, T is the transition vector and G is the gender vector. In the original SMPL paper, SMPL requires 72 pose vectors. In this case, the ground truth only contains 63 ground truth pose vectors. So pose vector has been reduced to 63. Also, SMPL requires 10 betas, and the ground truth dataset provided 16 vectors of betas. The first 10 betas have been chosen for training.

3 Methods

This part is introducing the design of the model, starting from data processing to model design and evaluation technique.

3.1 Data loader

The dataset is separated into 5 categories which are RGB images, Radar raw Images XYZ, SMPL-X parameters, 3D ground truth human body meshes and camera Intrinsics, and Extrinsics parameters. The information contained in each category will be introduced in separate sections below.

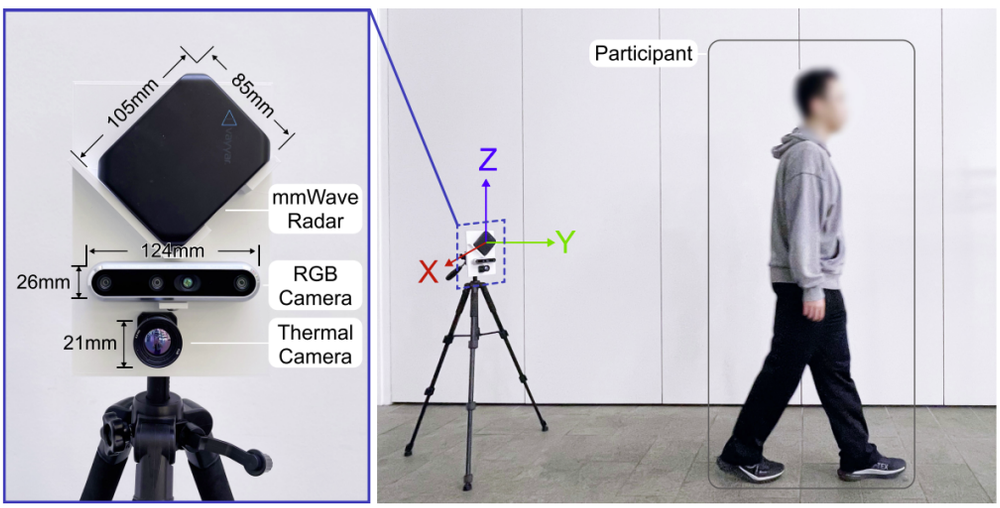

Figure 4: A diagram shows data collection process and position relationship between radar, camera and thermal camera

3.1.1 Radar data

The data of radar is collected by using the Vayyar vTrig Evaluation kit. This is a three-dimensional RF-based sensor and imaging system. The device outputs a Heatmap which is a 3D image matrix with size [121,111,31]. From the size, 121 and 111 are the width and length, 31 is the channel size and depth is default as 1. The radar data has passed a Fast Fourier transform on the vayyar, therefore, no further data cleaning process needs to be applied to the radar data.

3.1.2 3D Ground Truth Human Mesh and SMPL-X parameters

As the model is a supervised learning model, training requires ground truth data which gives the model a direction. The first step is to use the Vicon system to collect the point cloud of the human[15], Vicon will collect optical motion capture (MoCap) data and store them for further analysis. MoCap contains the point cloud data and pose information of a human in each frame. This is the information that Shape Optimization With

Mashes and Attributes(SOMA) requires[16]. SOMA is another system that analyzes MoCap information and results in them with SMPL-X parameters such as body shape, joint, rotation, and translation information. Furthermore, the most important is the human 3D mesh, generated using SOMA. Below is an example of sample data:

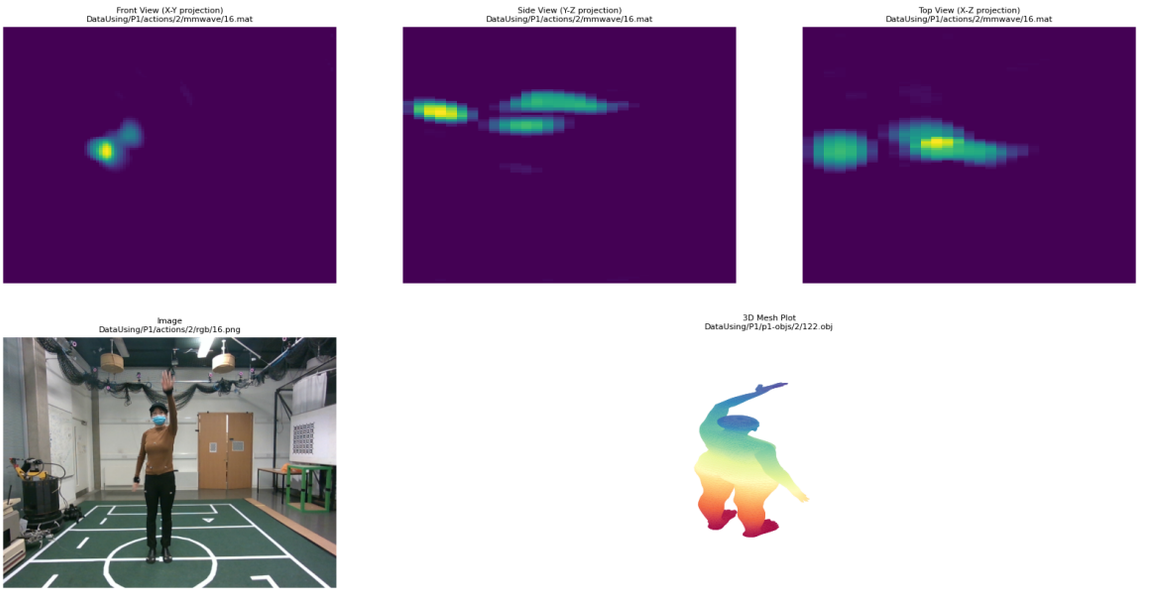

Figure 5: Data sample which contains raw radar image, RGB image and ground truth 3D mesh

The upper three images are the raw radar images from the side, front, and top view. The bottom left is the RGB image of the candidate, and the bottom right is the 3D ground truth human mesh of the candidate, which is demonstrated as point could.

3.1.3 Camera RGB image and calibrations

Visible RGB images are collected using a simple RGB camera as shown in 4. At the same time, when the camera takes an image of the candidate, radar takes a Heatmap of the candidate. To work on the same frame, the camera frame, Vicon frame, and radar frame need to be transferred to the same world frame. In this case, as the prediction data uses radar data as the input. Therefore, all other frames are converted into radar frames. Thecamera calibration is provided by the University of Edinburgh and they provided 4 files.Those are the calibration matrix of Vicon to the camera, the calibration matrix of radar to the camera, and the intrinsic of radar to the camera. The equation of how to calculate the calibration from Vicon to radar is shown below:

$$\mathbf{R}_{Vicon-to-Radar} = \mathbf{R}_{Vicon-to-Camera} \cdot \mathbf{R}^{-1}_{Radar-to-Camera}$$

$$\mathbf{T}_{vicon-to-radar} = \mathbf{R}^{T}_{vicon-to-radar}\left(\mathbf{T}_{vicon-to-Camera} - \mathbf{T}_{Radar-to-Camera}\right)$$

Use the equation above to calculate the rotation and translation matrix from the Viconframe to the radar frame. Finally, multiply the rotation matrix by to vision coordinate and plusthe translation matrix to get the coordinate under the radar frame shown below:

$$P_Radar = R_{Vicon-to-Radar} \cdot P_Vicon + T_{Vicon-to-Radar}$$

Data	Information
Radar raw image XYZ 1	Radar Heatmap, raw signal after FFT
vertices	Vertices from the ground truth 3D mesh
Faces	Faces from the ground truth 3D mesh
Joints	SMPL-X parameter, about human joints
Betas	SMPL-X parameter, Body shape, control the posture of the mian body
Body pose	SMPL-X parameter, joint rotation
trans	SMPL-X parameter, translation matrix for model movement
root orient	SMPL-X parameter, pelvis rotation
gender	The gender of the candidate

Table 1: Data and information that used for training

RF image data size	Point Cloud Data
[121,111,31]	[10475,3] vertices and [20908,3] faces.

Table 2: Comparing Radar data size and Cloud Point dataset size

The reason for using calibration is to reduce the vertices error which could be caused by systematic error.

3.1.4 Data Paring

Radar information is stored in mat type file and ground truth is stored in obj type file. To load data, first of all is to pair the data. Paring matched mat file, obj file and smpl-x parameters which stored in a json file.

3.1.5 Input data

All input data is converted into Tensor for better training. Because tensor performers better on GPU training. This part is summarising which type data has been input to the model for training. The table below is shown the data and information that the data contains:

3.2 Model

The model is designed into two parts, the first part is regression process to predict the SMPL-X parameters and the second part is to use the parameters to predict all the vertices and faces.

3.2.1 SMPL-X parameter prediction

This part is a regression task, starting with the structure of the model and moving to the detailed hyper-parameter of the model. The model is first designed with only a few layers of convolutional networks(CNN) because CNNs benefit from capturing the features and learning the relationship between input and output directly. These CNN layers aim to predict all the vertices that a 3D mesh needs, which is more than ten thousand vertices. Although that exceeded the capability of a small CNN model, this does prove the

work-ability of the CNN model. To improve that and avoid the model outputting the same result all the time, a new model has been designed to improve the performance by using SMPL-X as the final layer. SMPL-X only requests a few parameters as the input and predicts a 3D human mesh with 10475 vertices.

Figure 6: The overall working flow of the model. From input layer to output layer

The overall layer is shown in Figure 2, from the input 3D tensor layer to the output SMPX-X layer. The data first enters 3 CNN layers for down-scaling. The input channel is 31 channels and each CNN layer would output twice the output layer of the input layer, therefore, the output layer for each CNN layer is 64, 128, and 256 respectively. Because the radar data itself has lots of high-dimensional information, down-scaling could collect as many different dimensional features as possible. Down-sample also allows the model to reduce the probability of over-fitting and increase training efficiency.

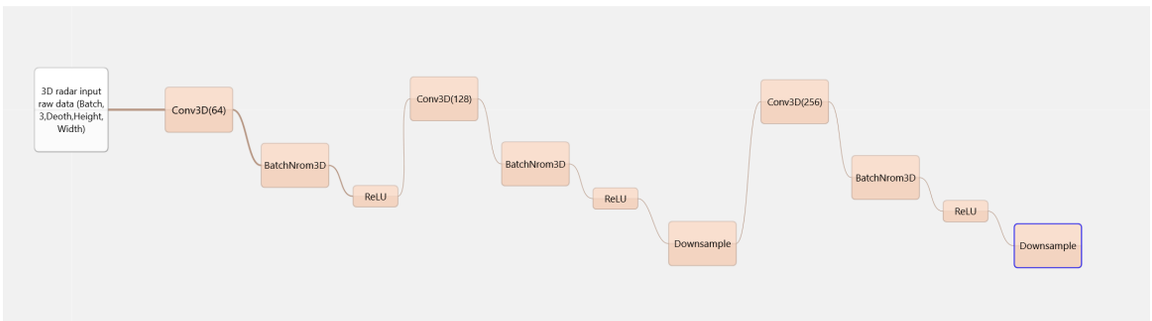

Figure 7: The detailed design of the CNN network

The design of the CNN layers is shown in Figure 3. Each layer uses 3 kernels and the ReLU activation function, but the second layer and the third layer have stride 2 for down-sampling. Each layer is followed by a batch normalization to speed up the training process and normalize the output to increase stability.
After CNN layers are the FPN layer with an attention mechanism. CNN layers have collected as many features as possible, and radar data included some background noise. Therefore, a technique is required to filter the noise and combine the high-dimensional features together. That is why using FPN layers is applied to the model in this project. Although it has shown great performance on feature selection and enhanced the correlation of the features, noise does affect a lot to the performance. So attention mechanism is added to the output of the FPN layer to balance the weight of different key features and reduce the effect of the noise. The reason for only using one layer of CNN and attention mechanism is to save training time and computational resources, Also, one layer of FPN performs as good as multiple FPN layers in this model.
After all these calculations, the size of the matrix is huge and out of the maximum capability of the GPU memory. To save time and reduce the waste of resources, originally, the model used max-pooling the keep the most important information. As a result, max pooling ignored some essential key features and caused difficulty in learning the joint, therefore, this model is using a 1X1 CNN layer instead. Finally, all features passed into the full connection layer for final prediction and output 5 different categories of

the SMPL-X model parameters. These parameters are betas, pose body, root orientation, translation, and gender.
The size of betas is 10, pose body is 62, root orient and translation is 3 for each and gender is 1. In total, those parameters are predicted separately as 5 matrices, not 79 parameters in total. The reason behind that is due to the different sizes of the parameters should apply different weights on it. This will be discussed in more detail in the loss function part.

3.2.2 SMPL-X parameter prediction

SMPL-X requires 166 parameters in total to predict high-quality human 3D mesh. In this project, the model only predicted 79 parameters. The missing parameters are left and right hands, and head features. Those are not included in this human 3D mesh. This project is mostly focused on human body pose rather than focusing on facial emotion. Therefore, the pre-training SMPL-X model has selected the headlock version, which does not require parameters of the head.
Before entering the parameters into the SMPL-X model, the model needs to select the correct model with the correct gender. In this project, 3 different models are prepared for selection. Gender prediction using 0 represent female and 1 represent male. When predicting the gender, the result could be around 0.5 which shows a not-too-specific gender feature, in this case, the model could choose SMPL-X neutral for this kind of problem. Human 3D mesh faces are using the pre-loaded SPML-X face with correspond gender.

3.2.3 Loss function

The loss function is the part to control the gradient descent. The model has predicted betas, body pose, root orientation, translation, gender, and vertices, therefore, all of these parameters are included in loss function calculations. The loss function uses using L1 loss function for all the parameters except gender prediction. And sum all the parameters together to calculate the final loss. Gender is using the L2 loss function. Each parameter loss function would have a weight to control the focus of the model. The equation is shown below:

$$\begin{aligned}\text{Loss Function} = {} & a \cdot |\text{Betas} - \hat{\text{Betas}}| + b \cdot |\text{pose} - \hat{\text{pose}}| + \\ & c \cdot |\text{orientation} - \hat{\text{orientation}}| + d \cdot |\text{translation} - \hat{\text{translation}}| + \\ & e \cdot |\text{vertices} - \hat{\text{vertices}}| + f * (\text{gender} - \hat{\text{gender}})^2\end{aligned}$$

Using the L1 function is because type 1 error uses absolute value and is not sensitive to outliers, which is suitable for regression tasks. Gender prediction is a classification task, therefore using type 2 error, which is sensitive to the outliers. The weight of each parameters will be adjusted by using Bayesian Optimization, and this will be explained in later section.

3.2.4 Other model

As mentioned on the previous page, in this project, the bottom half of the candidate moves less than the top half. This situation results in more weight needed to add on joint and body poses prediction than other parameters prediction. Therefore, to find the most suitable weight that could perform the best on the loss function, Bayesian Optimization was applied to the model and tried to find the minimum loss value during the duration

of the training. The iteration has been set to 15 epochs as a loop, and the initial weight of vertices is lowest at 0.001. That is because in total more than 10475 vertices are predicted, and the sum of the error will be a huge number compared with other parameters loss. Therefore, the initial weight of vertices has been set to 0.001. During some testing training, the top half of the body seems to have the problem. Especially the elbow. The elbow seems to be twisted in a different direction than what it should be. Details will be shown in the result section. To avoid that from happening, body pose and betas have set the initial weight to 2. Lastly, considering over fitting and local minimum, the optimiser of the model is using Adaptive Moment Estimation(Adam). This optimizer calculates the motion from historical gradient information to automatically adjust the learning rate for each parameter individually. Dropout is considered to prevent the model from overfitting as it randomly asks the model to forget some key feature, but for regression tasks, this could cause the learning is not continuous, Which could affect the stability of the model.
Lastly, this model is trained using two 4090 graphic card on a Linux server. Each iteration will take 2 days for training.

3.2.5 Evaluations

Calculate metrics for evaluations: Using the sum of the absolute error between prediction and ground truth value then using L2 normalisation to calculate the error. The equations of 5 Metrics are shown below.
1. average vertex error of all testing frame.

$$\text{vertex\_error} = \frac{1}{N}\sum_{i=1}^{N} \|\text{pred\_vertices}_i - \text{gt\_vertices}_i\|_2$$

2. average joint localization error

$$\text{joint\_error} = \frac{1}{M}\sum_{i=1}^{M} \|\text{pred\_joints}_i - \text{gt\_joints}_i\|_2$$

3. average joint rotation error

$$\text{rotation\_error} = \frac{1}{M}\sum_{i=1}^{M} \|\text{pred\_rotations}_i - \text{gt\_rotations}_i\|_F$$

4. mesh localization error

$$\text{mesh\_error} = \|\text{pred\_trans} - \text{gt\_trans}\|_2$$

5. gender prediction accuracy

$$\text{accuracy} = \frac{\text{number of correct predictions}}{\text{total predictions}}$$

4 Results

4.1 Data sample

This section shows some data samples. Figure 5 is the one sample of the data used for training. The top three photos are the raw radar XYZ images, from left to right representing the front view, side view, and top view. The light point is the position of the candidate and their arms in this sample specifically. This project aims to predict the bottom right 3D human mesh from those light points.

4.2 Data Size

In total, 97092 pairs of data are used in this project. Those are the data from 9 candidates. Within those 97092 pairs of data, 67965 pairs of data were used for training, 9709 pairs of data were used for validation and 29127 pairs of data were used for testing. Each candidate is provided with 50 poses with around 350 frames. The total number of parameters is 36652755, the most parameters come from FPN layers, 1769472 parameters.

4.3 Camera calibration

Because the machine learning model could not learn the extrinsic parameters from the data directly, camera calibration was applied to the Vicon data and convert them into the radar frame. To confirm the camera calibration accuracy and performance. Vicon point cloud is projected on matched RGB images. The result of the projection is shown below in figure 8 and 9.

Compared with the point cloud projection and RGB image, the Point cloud does not exactly cover the candidate, which shows that the extrinsic is not exactly right which leaves some small systematic error. Machine learning could not learn the extrinsic. All of these factors could have a negative effect on the final results. The projection seems to be in front of the real candidate in the image, the problem might be the depth. Also, not all the point clouds and projections are fitting such as figure 10. Human Mesh seems to be off the page by a huge distance. Those are the outliers that may confuse the model while training. Although some of the outliers are removed from the training set, some may still remain in the training dataset.

4.4 loss iteration

In this project, both single-head self-attention and multi-head self-attention are experimented with. Plotting figure 11 and 12 are the loss change for single-head self-attention and figure 13 and 14 are the loss change for multi-head self-attention. Overall, both attention methods have shown a good performance in training the model. Loss decreases generally and levels off in the end which means models are well trained. Those two plots are the second and third iterations from Bayesian Optimization. Multi-head seems to show a better performance at the same iterations as in the third iteration, the validation loss of multi-head attention does not get over the training loss. That is because the input of the model is a list of tensor data and the output is a list of data as well, multi-head attention could add the weight to different keys separately rather than giving a single global weight. This could explain why multi-head self-attention performs better

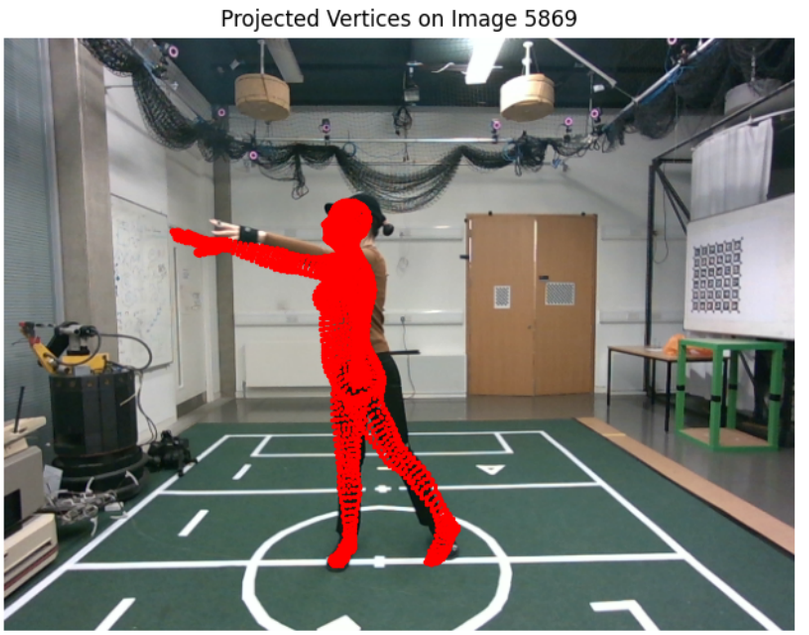

Figure 8: Example 1 of projecting 3D mesh on image

in this regression task. Also, multi-head self-attention improves the right arm prediction. That might because some keys added weight to the upper body for more secure predictions.When calculating the loss, betas, and poses times by three increased the loss value, therefore, the loss value does not decrease to one. No over-fitting occurs in the training as no huge fluctuation or unexpected increase trend.

4.5 3D mesh Prediction

Figure 15,16,17,18,19, are the 3D mesh prediction using multi-head attention method. Figure 20,21,22,23,24, are the 3D mesh prediction of single head attention method.
To have a clear evaluation of the 3D meshes the plotting has top, front, and side views. From the plotting, both single-head and multi-head methods are doing well on general pose prediction. Body pose and position are generally correct. However, the single-head method has more errors in joint prediction. In figure 15 right arm should be backward rather than forward, but other poses are in an acceptable range. Figure 16 prediction bends the knees whereas ground truth does not. Figure 17 the arm is perpendicular to the body rather than pointing forwards. All of the multi-head attention results show a single joint problem, most of the problems are related to the right arm elbow joint

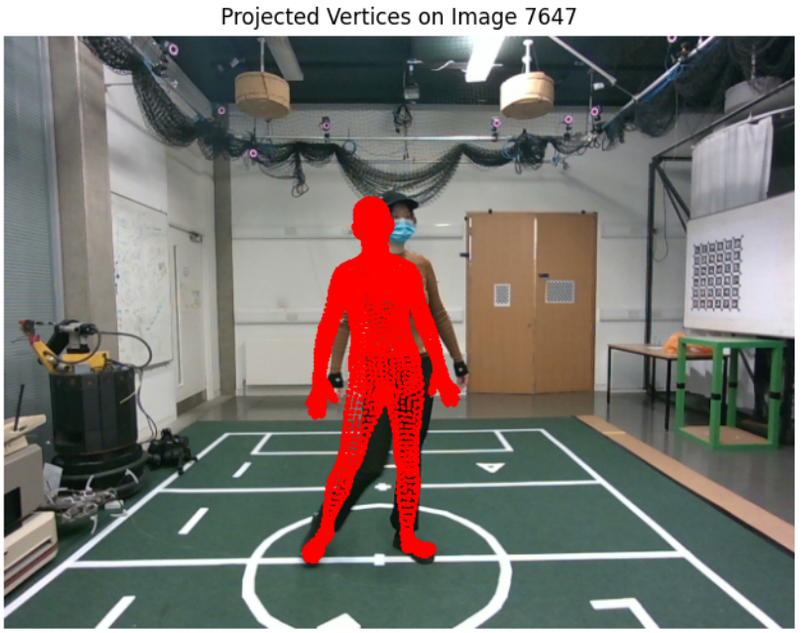

Figure 9: Example 2 of projecting 3D mesh on image

prediction. This problem could be caused by several reasons.
First, radar data is given limited information about the arms, from the example data. right arm information may have been combined with body information. Second, the model may not be able to identify a complex pose, such as twisting the arm. Compared with the vertices on the arm, most of the vertices are on the leg and main body. Therefore loss function may focus more on body prediction rather than arm prediction. Third, Within all 50 poses, the arm moves more than the legs. Too much data contains arm movements which confuse the model.
Single head self-attention model has the same problem but even worse. In Figure21 right arm has been extended, and the length of the arm is incorrect. In Figure 23 Both arms have an error, arms should open rather than close to the body. In figure 24 Both knees and arms have the problems. Therefore, the result proves that multi-head attention performers are better.

4.6 Metrics calculation

Tables 3 and 4, show multi-head self-attention performs with fewer errors. The model achieved an average vertex error of 12 centimeters (cm) and the average joint localization

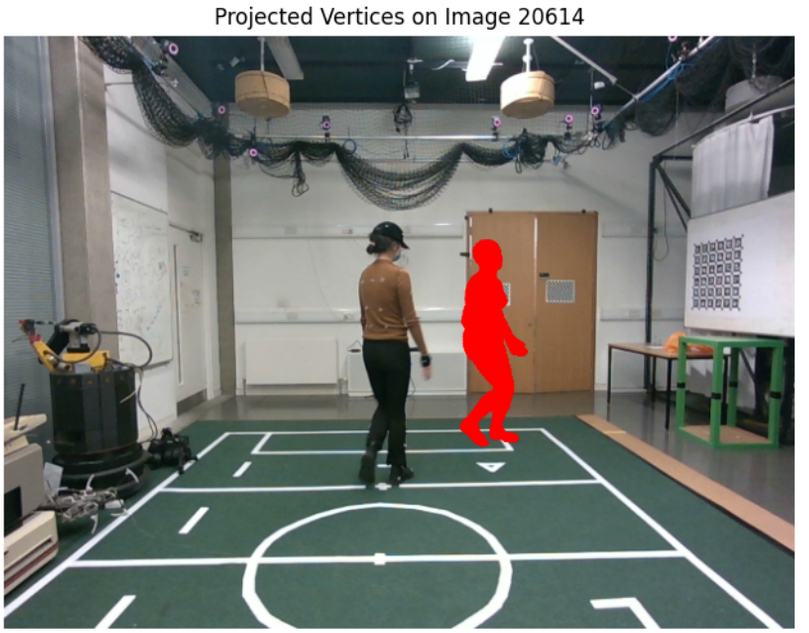

Figure 10: Example 3 of projecting 3D mesh on image

error was 14 centimeters. Reflected on the 3D mesh figures, the majority of the vertices are predicted correctly and the errors may be from the right arms. In total 10457 vertices are predicted, and a small change of pose could result in a huge error in vertices and joint localization. The mesh localization error was 40 centimeters. This may caused by the calibrations because the current result is not calibrated and machine learning algorithms could not learn the calibration relations. Therefore, a huge error could cause the whole human 3D mesh to shift to one side. Gender classification performers well with 99.8 percentage, so all 3D meshes should have a correct gender features. Lastly, one key point needs to be mentioned. The model shows a low performance with untrained human poses.

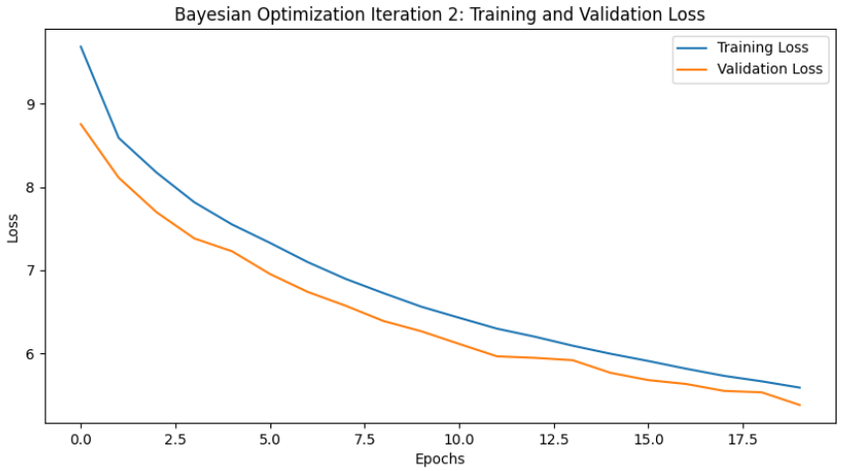

Figure 11: Loss function plot of the single head self- attention iteration 2

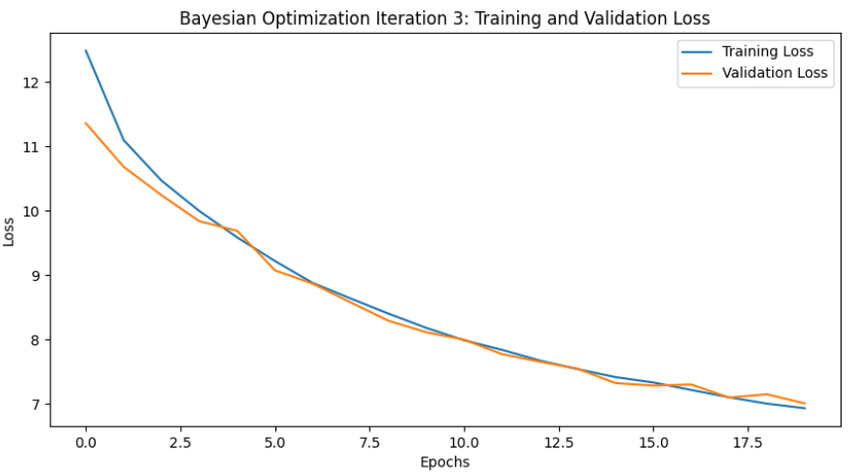

Figure 12: Loss function plot of the single head self- attention iteration 3

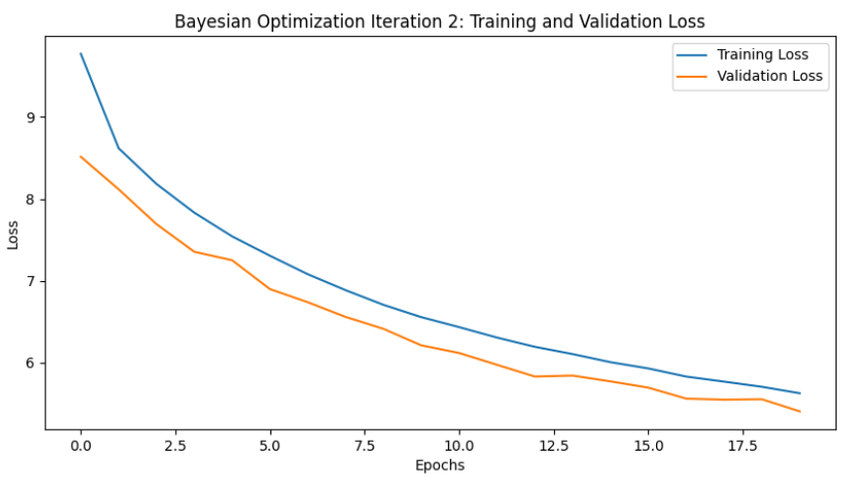

Figure 13: Loss function plot of the multi-head self- attention iteration 2

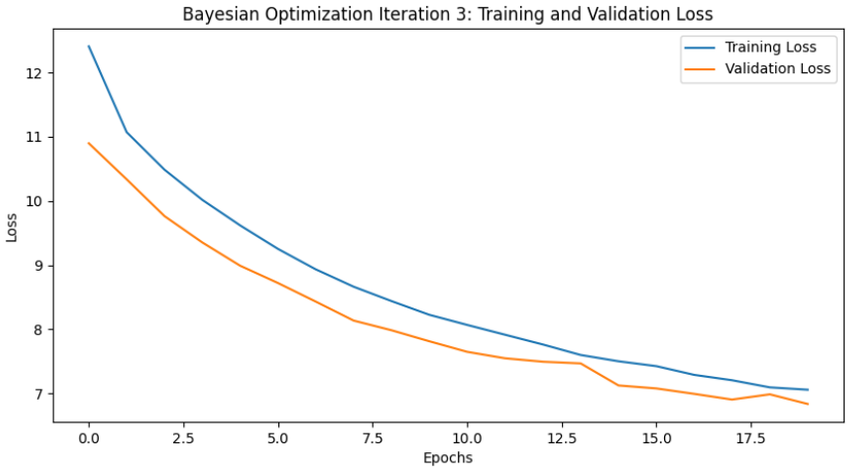

Figure 14: Loss function plot of the multi-head self- attention iteration 3

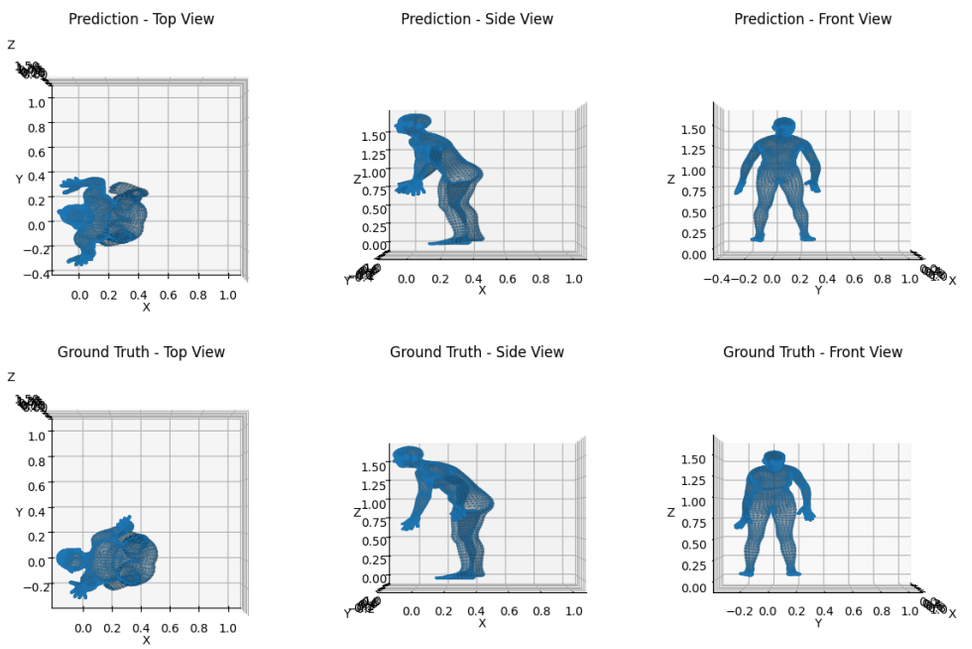

Figure 15: Multi-head self-attention 3D mesh plot testing sample 1

Metrics Error (without calibration)	value	mmMesh baseline
average vertex error 1	12 cm	2.47 cm
average joint localization error	14 cm	2.18cm
average joint rotation error	1.13 degree	3.8 degree
mesh localization error	40 cm	1.27 cm
gender prediction accuracy	99.6 percent	99.8 percent

Table 3: Multi-head 3D mesh prediction error Metrics calculation

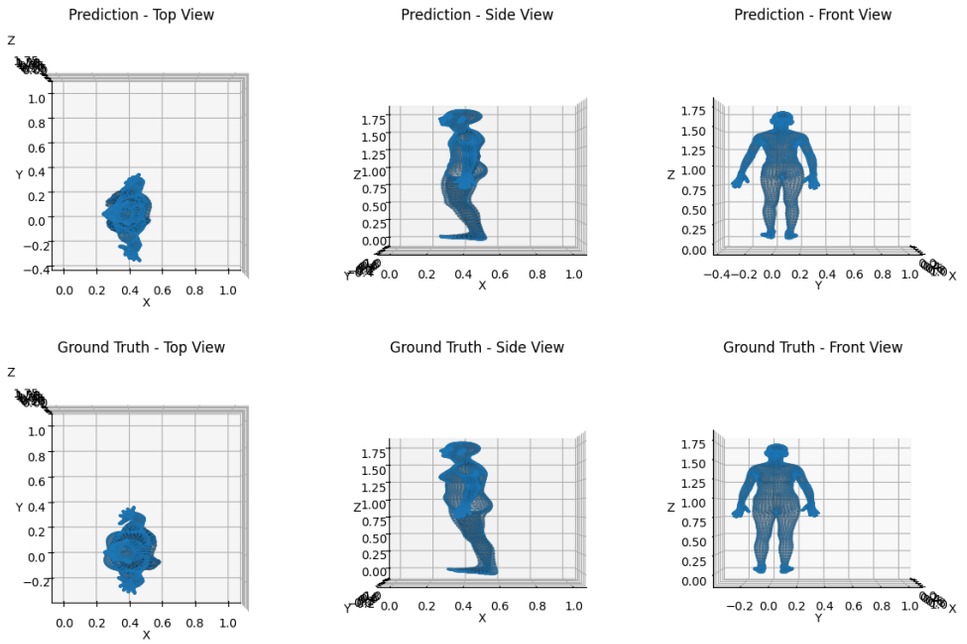

Figure 16: Multi-head self-attention 3D mesh plot testing sample 2

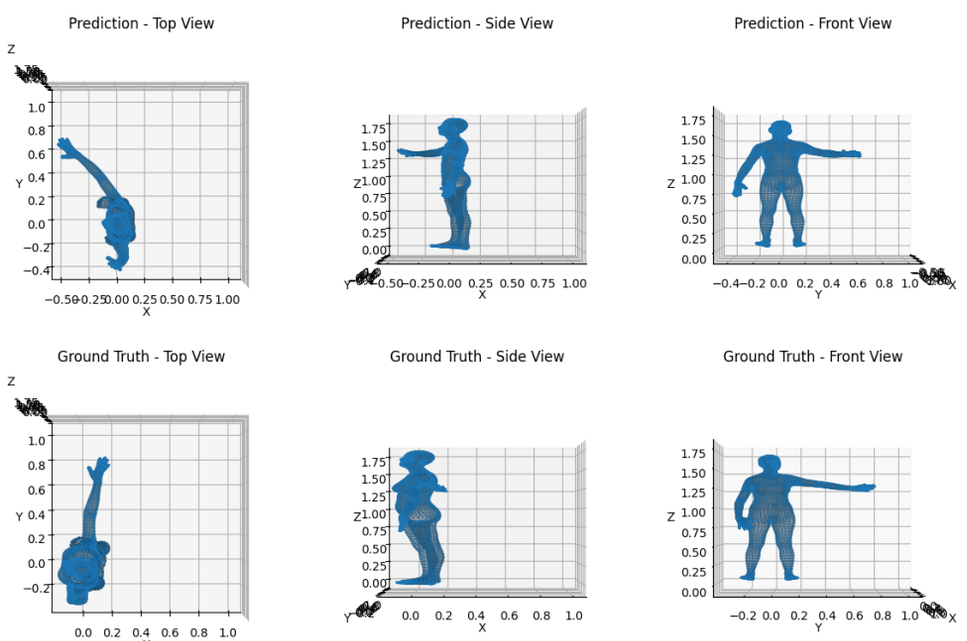

Figure 17: Multi-head self-attention 3D mesh plot testing sample 3

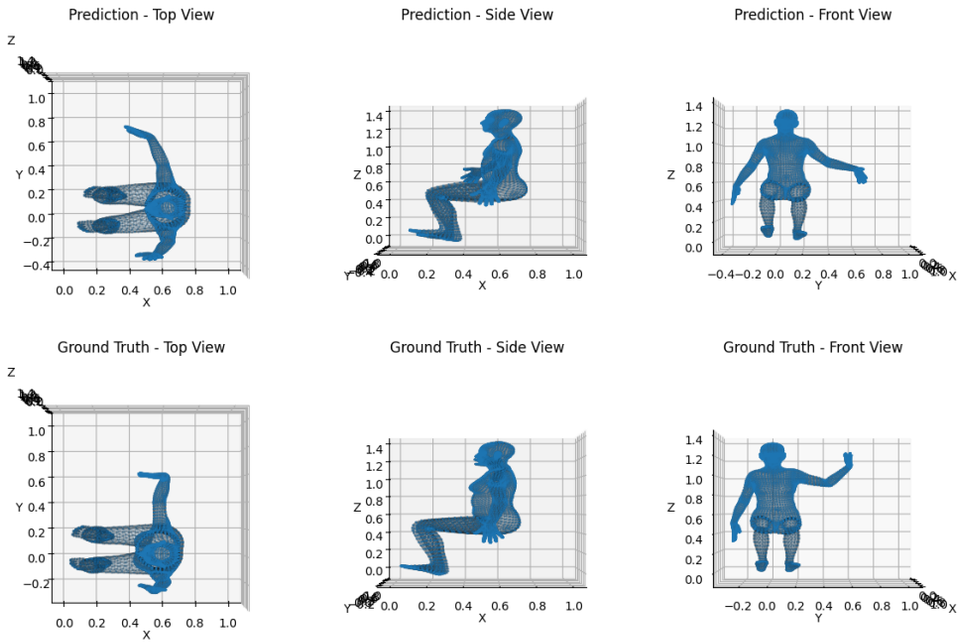

Figure 18: Multi-head self-attention 3D mesh plot testing sample 4

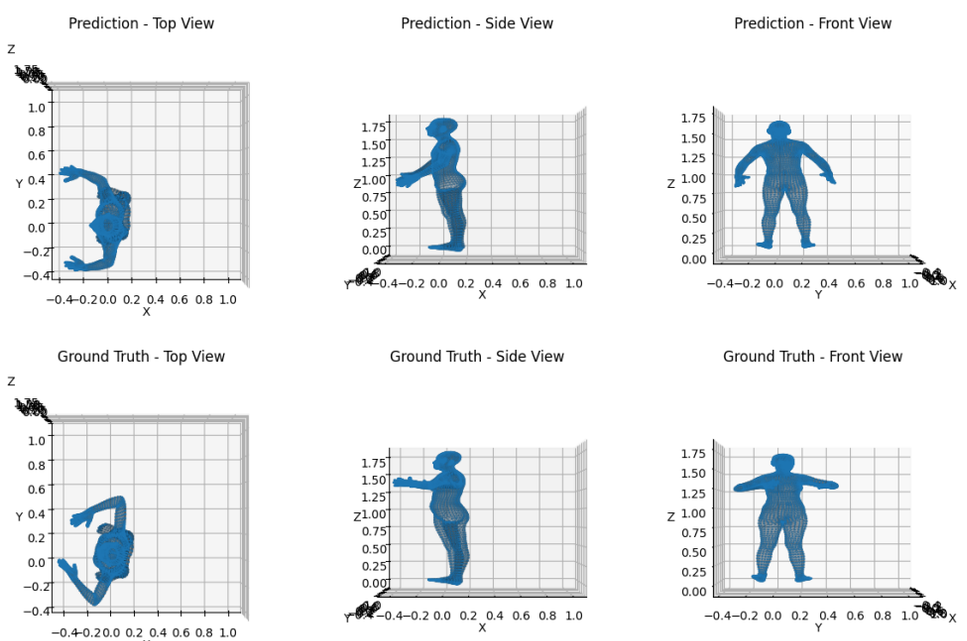

Figure 19: Multi-head self-attention 3D mesh plot testing sample 5

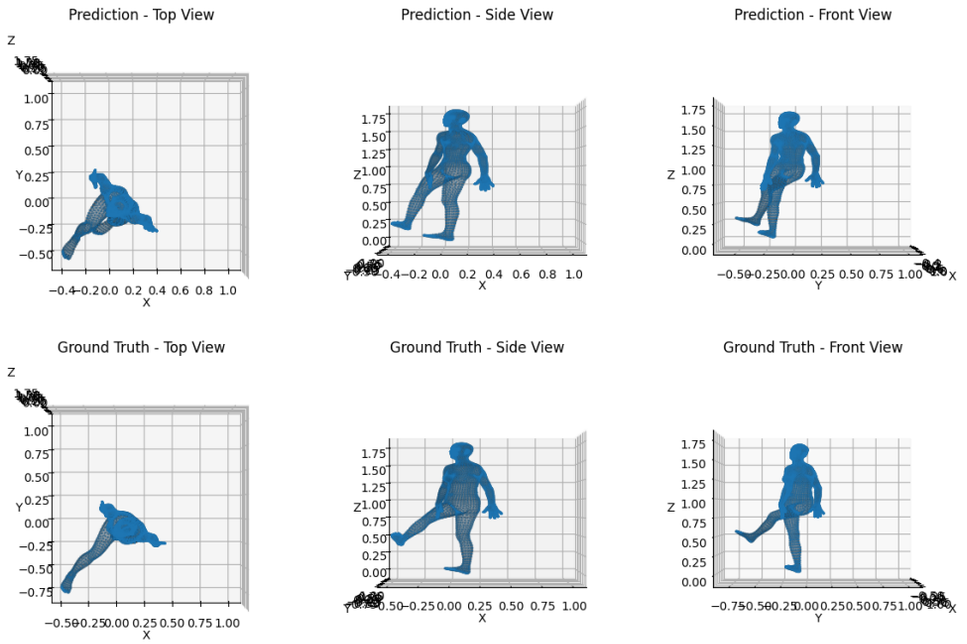

Figure 20: Single-head self-attention 3D mesh plot testing sample 1

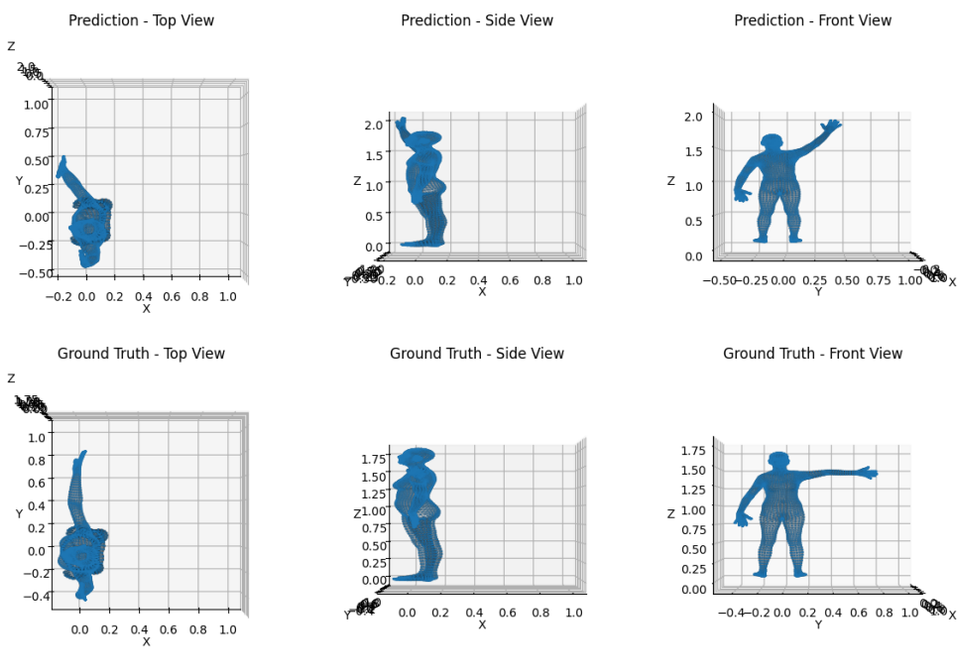

Figure 21: Single-head self-attention 3D mesh plot testing sample 2

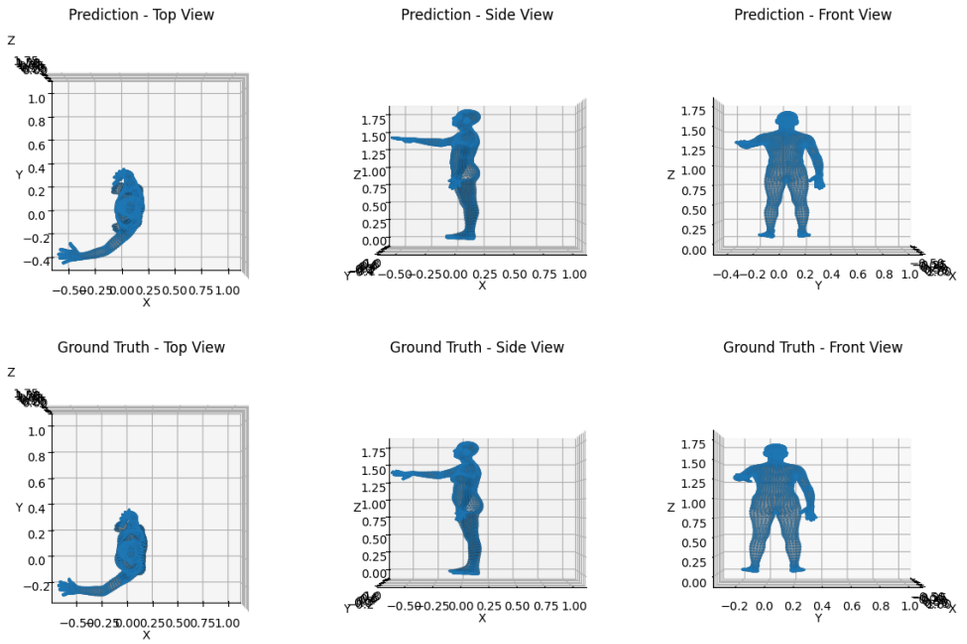

Figure 22: Single-head self-attention 3D mesh plot testing sample 3

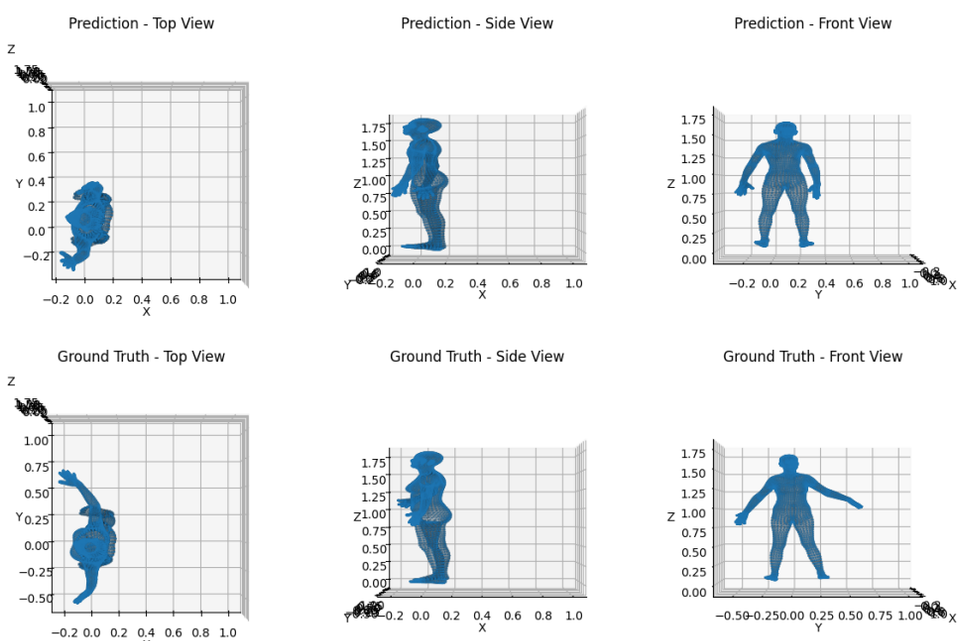

Figure 23: Single-head self-attention 3D mesh plot testing sample 4

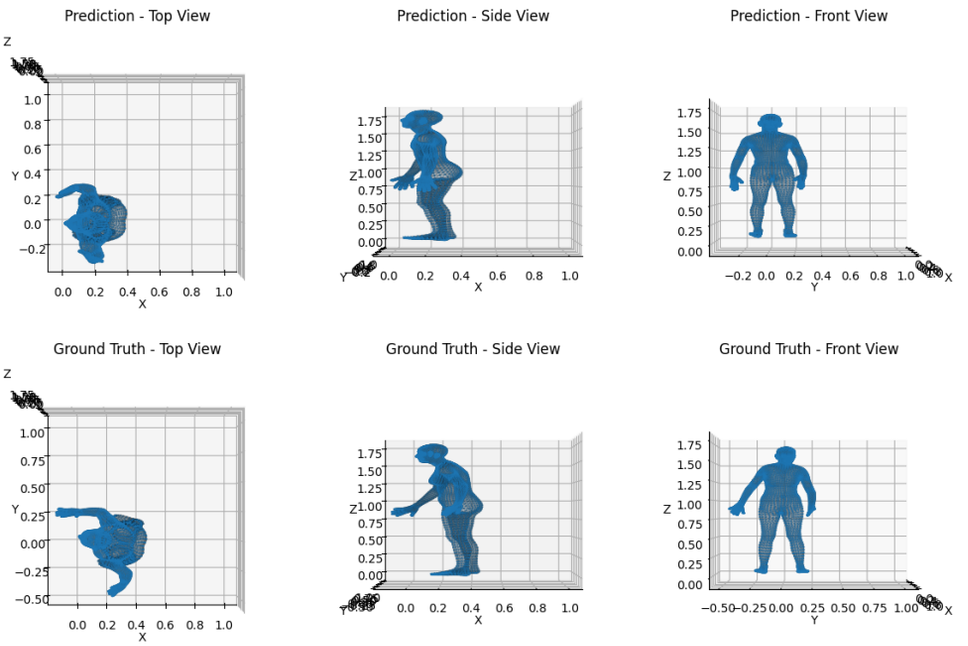

Figure 24: Single-head self-attention 3D mesh plot testing sample 5

Metrics Error (without calibration)	**value**	**mmMesh baseline**
average vertex error 1	15 cm	2.47 cm
average joint localization error	20 cm	2.18cm
average joint rotation error	3.4 degree	3.8 degree
mesh localization error	50 cm	1.27 cm
gender prediction accuracy	99.6 percent	99.8 percent

Table 4: Single-head 3D mesh prediction error Metrics calculation

5 Conclusions

This research provides a new neural network model, SimpleCFA, aimed at reconstructing 3D human body meshes using only millimeter-wave radar data. The model benefits from working under poor visibility conditions. The model is structured into two parts, the SMPL-X parameters regression task and the SMPL-X human 3D mesh prediction task. SMPL-X parameters regression task is constructed with three CNN layers, a feature pyramid network, and a multi-head self-attention mechanism to extract and process critical features from radar data. Then SMPL-X received the result from the regression part to generate a 3D human mesh with gender features. As a result, SimpleCFA is promising with an average vertex error of 12 cm, an average joint localization error of 14 cm, and a joint rotation error of 1.13 degrees. Furthermore, the model achieved a gender prediction accuracy of 99.8 percent, highlighting its robustness in classifying gender features based on radar data.

SimpleCFA model has provided an idea that millimeter-wave radar data can be effectively used to reconstruct 3D human meshes without optical information such as RGB images and LiDar. The integration of advanced neural network architectures, such as the multi-head self-attention mechanism, significantly enhanced the model's performance, making it a viable alternative in scenarios where optical data is unreliable or unavailable. Further on that, radar information could pass through cloths and walls where LiDar and images could be blocked.

However, The model still has some problems due to the time limitation that could not be finished. Camera calibration is one of the most important parts that needs to be considered. Although the average mesh localization is 40 cm, if the calibration could be improved, the accuracy would improve dramatically. Right elbow or arm prediction is still the problem. After using self-attention and the FPN layer, the model is still confused with the right arm. In this case, training data may mix some information which causes CNN and FPN to separate those information.

Looking forward, future research could explore the inclusion of temporal data to better capture dynamic poses and further refine radar data processing techniques to minimize localization errors. Datasets are collected in the same light condition and the same environment. SimpleCFA model performed well under the same condition with a clear background but lacked experiment in complex backgrounds or different light condition. Within those conditions, more feature selection layers may be required for feature selection and background noise removal. Furthermore, another state-of-the-art method has achieved multi-people 3D reconstruction by using a Region Proposal Network for human separations. This is a more realistic problem in the auto-driving field. Also, the model should designed with real-time 3D reconstruction, which means the model needs to have a layer to learn time stamp-related information. Long - Short-Term Memory is the solution provided by the mesh method. That could be added in future research.

In conclusion, this research successfully demonstrated the potential of using millimeter-wave radar data for 3D human mesh reconstruction, offering a novel approach that could significantly impact fields ranging from motion capture to privacy-sensitive environments.

References

[1] V. S. Kulkarni, R. B. Trivedi, and S. R. Goyal, "A novel technique for converting images from 2d to 3d using deep neural networks," in *2023 International Conference on Innovative Data Communication Technologies and Application (ICIDCA)*, 2023, pp. 521–530.

[2] Y. Li, J. Deng, Y. Zhang, J. Ji, H. Li, and Y. Zhang, "ezfusion: A close look at the integration of lidar, millimeter-wave radar, and camera for accurate 3d object detection and tracking," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 182–11 189, 2022.

[3] S. K. Dwivedi, Y. Sun, P. Patel, Y. Feng, and M. J. Black, "Tokenhmr: Advancing human mesh recovery with a tokenized pose representation," 2024. [Online]. Available: https://arxiv.org/abs/2404.16752

[4] A. M. Wallace, A. Halimi, and G. S. Buller, "Full waveform lidar for adverse weather conditions," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 7, pp. 7064–7077, 2020.

[5] S. Plosz, A. Maccarone, S. McLaughlin, G. S. Buller, and A. Halimi, "Real-time reconstruction of 3d videos from single-photon lidar data in the presence of obscurants," *IEEE Transactions on Computational Imaging*, vol. 9, pp. 106–119, 2023.

[6] G. W. Kim, S. W. Lee, H. Y. Son, and K. W. Choi, "A study on 3d human pose estimation using through-wall ir-uwb radar and transformer," *IEEE Access*, vol. 11, pp. 15 082–15 095, 2023.

[7] M. Zhao, Y. Liu, A. Raghu, H. Zhao, T. Li, A. Torralba, and D. Katabi, "Through-wall human mesh recovery using radio signals," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 10 112–10 121.

[8] H. Xue, Y. Ju, C. Miao, Y. Wang, S. Wang, A. Zhang, and L. Su, "mmmesh: towards 3d real-time dynamic human mesh construction using millimeter-wave," in *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 269–282. [Online]. Available: https://doi.org/10.1145/3458864.3467679

[9] M. Zhao, Y. Tian, H. Zhao, M. A. Alsheikh, T. Li, R. Hristov, Z. Kabelac, D. Katabi, and A. Torralba, "Rf-based 3d skeletons," in *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, ser. SIGCOMM '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 267–281. [Online]. Available: https://doi.org/10.1145/3230543.3230579

[10] R. Chauhan, K. K. Ghanshala, and R. Joshi, "Convolutional neural network (cnn) for image detection and recognition," in *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*, 2018, pp. 278–282.

[11] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *2017 International Conference on Engineering and Technology (ICET)*, 2017, pp. 1–6.

[12] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," 2017. [Online]. Available: https://arxiv.org/abs/1612.03144

[13] Y. Jiang, Q. Liao, Z. Wang, X. Lin, Z. Lu, Y. Zhao, H. Wei, J. Ye, Y. Zhang, and Z. Shao, "Smplx-lite: A realistic and drivable avatar benchmark with rich geometry and texture annotations," 2024.

[14] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. A. Osman, D. Tzionas, and M. J. Black, "Expressive body capture: 3d hands, face, and body from a single image," 2019.

[15] Y. Bai, H. Hu, Y. Li, C. Zhao, L. Luo, and R. Wang, "Research methods for human activity space based on vicon motion capture system," in *2017 5th International Conference on Enterprise Systems (ES)*, 2017, pp. 202–206.

[16] N. Ghorbani and M. J. Black, "Soma: Solving optical marker-based mocap automatically," 2021.

A Extra Material

please add as many appendices you need