mmWaveFlow: Unified Enhancement and Generation of mmWave Human Point Clouds

Supplementary Material

A. Additional Experimental Evaluations

A.1. Complexity and efficiency

We report the model complexity and training/inference efficiency of mmWaveFlow and the comparison methods in Tab. 5. The number of model parameters and the FLOPs (Floating Point Operations) of a single forward propagation are computed by the calflops library in Python. Training time is reported as the average time per batch with a batch size of 64. Inference time is reported for all models with a batch size of 64 and 50 sampling steps. For a fair comparison with the baselines, we report only the single-direction inference time of mmWaveFlow, corresponding to generating mmWave point clouds from dense point clouds. The mmPoint model directly regresses the outputs in a single forward pass without multi-step sampling, resulting in low inference time. However, its performance on the evaluated tasks remains relatively limited. In addition, training mmWaveFlow without gradient surgery further reduces the training time to 0.33 s per batch. Overall, mmWaveFlow achieves strong generation performance while maintaining competitive inference efficiency.

A.2. Generation of mmWave point clouds

We further evaluate the impact of the generated point clouds on two additional behavior recognition methods that operate on point clouds, PSTNet [1] and Set Transformer [2]. As shown in Tab. 6, augmenting the training set with our generated point clouds can improve the performance of both methods, demonstrating that the proposed data generation strategy is effective and broadly beneficial for human action recognition from mmWave point clouds.

A.3. Additional ablation study

Training Strategies. We conduct an ablation study of the two training strategies used in our framework. We treat the entire training process as a multi-task learning problem. To mitigate potential gradient conflicts arising from different loss terms, we adopt a gradient surgery method [3] to reduce interference between tasks. As shown in Tab. 7, applying gradient surgery improves mmWaveFlow on 3 out of 4 evaluation metrics. In addition, we empirically observe that continuously updating the VAE causes the latent tokens produced by the encoder to exhibit small perturbations, which hampers the convergence of the OA-Flow network. To stabilize training, we freeze the VAE parameters once the VAE has approximately converged. By comparing performance results with and without freezing VAE parameters, we find that this strategy further improves the performance of mmWaveFlow.

Table 5. Comparison of model complexity and efficiency metrics across different models.

Model	#FLOPs	#Params	Training Time	Inference Time
mmPoint	5.6 (G)	8.78 (M)	0.72 s	0.05 s
RadarHD	1.83 (G)	17.49 (M)	0.10 s	1.51 s
RadarDiff	3.43 (G)	36.42 (M)	0.18 s	2.75 s
LiDiff	0.55 (G)	32.70 (M)	0.39 s	9.31 s
Tiger	96.2 (G)	70.04 (M)	1.67 s	23.64 s
mmWaveFlow	5.17 (G)	179.94 (M)	0.78 s	1.36 s

Table 6. Action recognition on MM-Fi dataset (continued from Tab. 3.

Method	Training Set	Training Epoch	Accuracy
Set Transformer	S_{DT}	20	0.557
	S_{FT}	20	0.473
	S_{FT}+$\hat{S}_{DT}$	10	0.603
PSTNet	S_{DT}	20	0.672
	S_{FT}	20	0.659
	S_{FT}+$\hat{S}_{DT}$	10	0.742

Table 7. Ablation study on training strategies.

Model	D2M		M2D	
	CD	EMD	CD	EMD
mmWaveFlow	2.69	8.49	0.75	3.69
w/o Gradient Surgery	3.27	8.16	0.79	3.74
w/o Freeze VAE	3.36	8.63	0.83	3.86

A.4. Performance across different sparsity levels

The three datasets roughly cover different sparsity levels. Taking the MM-Fi dataset as an example, even after multi-frame aggregation, the training samples have only an average of 138.51 points.

Since many points are repeated across adjacent frames, each aggregated sample contains only 43.73 unique points on average, which is quite sparse.

We extract extremely sparse samples (with 10–50 points) and relatively sparse samples (with 50–100 points) from the test set of MM-Fi dataset for separate evaluation. The results are shown in Tab. 8.

Overall, the sparser the data, the worse the performance of mmWaveFlow. Extremely sparse data may lack sufficient

Table 8. Performence of mmWaveFlow on extremely sparse data.

Range of Sample Point Counts	Number of Total Samples	Average Number of Unique Points per Sample	D2M		M2D	
			CD	EMD	CD	EMD
(10,50]	9908	12.34	4.71	9.19	0.89	4.06
(50,100]	19653	24.97	3.11	8.76	0.82	3.87

discriminative information. This issue may require hardware improvements, not just algorithmic solutions alone.

B. Discussion

In this work, we propose mmWaveFlow, a unified flow-matching framework for the generation and enhancement of mmWave human point clouds. By jointly modeling these two tasks within a single model, mmWaveFlow can help mitigate two fundamental limitations of existing mmWave human point cloud data, namely the severe sparsity and low quality of mmWave point clouds and the limited scale of available datasets.

We validate the effectiveness of mmWaveFlow on two representative downstream tasks. For human mesh recovery, enhancing the mmWave point clouds with mmWaveFlow improves the accuracy of human mesh recovery. For behavior recognition, using mmWaveFlow to generate additional mmWave point clouds for data augmentation leads to improved recognition performance.

Although mmWaveFlow achieves promising results, it remains an initial step toward bidirectional translation between sparse mmWave and dense human point clouds. Our study primarily focuses on addressing semantic alignment and path-crossing issues in flow-matching training. However, several factors still limit performance.

In particular, experiments suggest that the core challenge lies in modeling the transformation between dense and extremely sparse mmWave observations, where severe sparsity leads to information loss and ambiguity. In addition, noisy outliers in mmWave data further complicate the evaluation of generated results. These findings indicate that more expressive representations, improved model architectures, and modality-specific designs—such as mmWave-tailored generative backbones—remain under-explored and offer promising directions for future research.

References

[1] Hehe Fan, Xin Yu, Yuhang Ding, Yi Yang, and Mohan Kankanhalli. Pstnet: Point spatio-temporal convolution on point cloud sequences. In *International Conference on Learning Representations*, 2021. 1

[2] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International Conference on Machine Learning*, pages 3744–3753. PMLR, 2019. 1

[3] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in Neural Information Processing Systems*, 33:5824–5836, 2020. 1