

Channel-Spatial Attention Enhanced Few-Shot Radar-based Hand Gesture Recognition

Krishnapriya S.¹, Seshan Srirangarajan^{1,2}, and Sumantra Dutta Roy¹

¹Department of Electrical Engineering, Indian Institute of Technology Delhi, India

²Bharti School of Telecommunication Technology and Management, Indian Institute of Technology Delhi, India

Abstract—Radar-based hand gesture recognition is a promising modality for human computer interaction (HCI) due to its advantages over camera-based approaches. We propose a novel few-shot hand gesture recognition model, using range-Doppler images (RDI), that can generalize with a limited number of samples. The proposed model uses a prototypical network with a customized feature embedding network and channel-spatial attention to effectively capture the spatial and temporal features from the RDI frames. We carry out extensive experiments of the proposed prototypical network-based few shot framework on four public radar-based hand gesture datasets. The results show promising gesture recognition accuracy and good generalizability, validating the proposed framework and its applicability in real-world scenarios.

Index Terms—Range-Doppler imaging, few-shot learning, hand gesture recognition, prototypical networks, temporal convolutional networks, attention mechanisms.

I. INTRODUCTION

Dynamic hand gestures are widely used for non-contact human-computer interaction (HCI). Vision-based sensors, which are commonly employed for this purpose, are heavily influenced by lighting and environmental conditions and also raise privacy concerns as they capture visible biometric data such as faces, hands, and behavioral patterns. Surface electromyography (sEMG) sensors on the other hand, have to be placed on a subject's forearm which can be intrusive and less convenient. In contrast, radar-based sensors use electromagnetic (EM) waves to detect objects in their field of view, offering a non-contact interface, enhancing usability and accessibility (e.g., while driving). Radar sensors are more robust than vision-based sensors under challenging conditions such as low illumination, rain, and fog. Additionally, EM waves can penetrate certain materials, making them effective in scenarios with occlusion. Since radar sensors do not capture visual data or personally identifiable information, they are more suitable for applications with privacy concerns.

Training of hand gesture recognition models typically requires a large number of annotated samples. Few-shot (FS) learning [1], a promising meta-learning approach, can effectively learn and generalize from a much smaller set of training samples. This significantly reduces the effort required for data collection, annotation, and training time. It enables rapid deployment and updates of hand gesture recognition systems without the need for an extensive training phase. In this work, we use the radar range-Doppler images (RDI) which contain both spatial and temporal information. This necessitates a suitable framework to extract significant spatio-temporal features from the data stream. The contributions of this work are as follows:

- We propose a novel architecture, based on the general framework of prototypical few-shot learning, designed for RDI frames. We propose a domain-specific feature extraction module, which accounts for attention mechanism to obtain suitable spatio-temporal representation for the gesture recognition task.
- The proposed model entails a feature embedding network comprising a 3D convolutional neural network (C3D), a temporal convolutional network (TCN), and a channel-spatial attention (CSA) block. This combination of blocks uses attention on

specific spatio-temporal features, which enables the selection of the best prototype in the prototypical network.

- We demonstrate the performance of the proposed model through extensive experiments on *four* large publicly available datasets. We also present a detailed ablation study on different parts of the proposed architecture.

II. RELATED WORK

Few-shot meta-learning [1] is a technique that helps a model to learn from minimal number of samples. Metric-learning is a sub-category of few-shot meta-learning that tries to minimize a distance function over the data samples. Popular metric learning techniques include Siamese networks [2], matching networks [3], relation networks [4], and prototypical networks [5]. Siamese networks compute the distance between the outputs of a pair of identical neural networks with shared weights. Koch *et al.* [2] use the ℓ_1 -distance for measuring image similarity. Matching networks [3] define a probability distribution over output labels using an attention kernel. The attention kernel computes the cosine similarity between the embeddings of support and query examples, and performs softmax normalization. The classifier output is a sum of the labels (one-hot encoded) of support samples, weighted by the attention kernel. Relation networks [4] replace the metric computation in prototypical networks with a CNN to compute the similarity score. Attention enhanced few-shot relation networks are explored in [6], [7].

Among metric few-shot meta-learning models, prototypical networks [5] are often simple, and give comparable performance to the other parameter-heavy models [4]. A traditional prototypical network has a CNN feature extractor. The prototype for each class in the support set is obtained by taking the average of the embeddings of the support set samples of that class. The similarity is measured as the squared Euclidean distance between the embedding of the query and each class prototype. The output probability over classes is through a softmax over the negative of the distance function.

Few-shot hand gesture recognition has been explored with different sensing modalities. In [8], few-shot learning has been used with sEMG

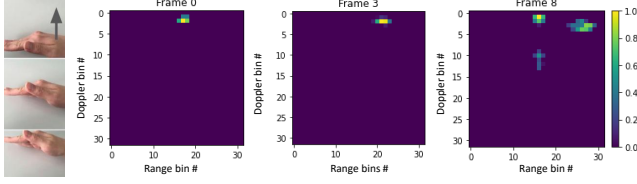


Fig. 1: Three significant RDI frames of the pull gesture from the Soli [19] dataset showing the intensity of the reflections from the hand as it moves along the range dimension.

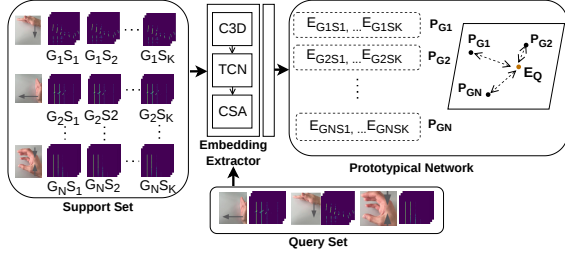


Fig. 2: Proposed few-shot prototypical network with a novel feature embedding network. Prototypes (P_{G_i}) are calculated as mean of the embeddings of the K samples in a gesture class G_i . Metric learning analyzes the distance between the N prototypes and the query sample embedding (E_Q) to determine its label.

signals. In [9], the authors use accelerometer data collected with wrist-worn devices, towards a self-supervised technique. Ma *et al.* [10] use a skeleton-based few-shot hand gesture recognition system using the coordinates of the finger joints, while Liu *et al.* [11] use hand gesture RGB data. Based on relation networks, Zeng *et al.* [12] propose a weighting network for micro-Doppler spectrogram of dynamic hand gestures. Mauro *et al.* [13] use a combination of range time maps (RTM), Doppler time maps (DTM), and angle time maps (ATM). In [14], the authors present ML-HGR-Net using both range-Doppler and angle-Doppler maps by fusing the 3D-CNN outputs.

Attention mechanisms have been adopted in most computer vision models due to their ability to improve feature extraction by the models. In [15], a convolutional block attention module (CBAM) refines intermediate features with two distinct channel and spatial modules. This module has also been adapted in various fields [16]–[18]. It is evident from the literature that there is a need to develop few-shot learning models for radar-based gesture recognition. Addressing this could lead to more robust and adaptable gesture recognition systems, especially in scenarios where data collection is challenging and real-time adaptation to new gestures is required.

III. FEW-SHOT PROTOTYPICAL FRAMEWORK

In this work, we propose a few-shot learning architecture using channel and spatial attention for gesture recognition based on RDI sequences (refer Fig. 1). RDI sequences are obtained by applying 2D-FFT on the radar ADC data. Fig. 2 shows the proposed model architecture. Support set and query set are given as inputs to the model. The model classifies an RDI sequence in the query set into one of the N gesture classes in the support set with the help of limited (K) samples available per gesture class. Let $G_n S_k$ represent the k^{th} sample of the n^{th} gesture class from the support set. The initial part of the model is an embedding extractor that includes sub-networks to

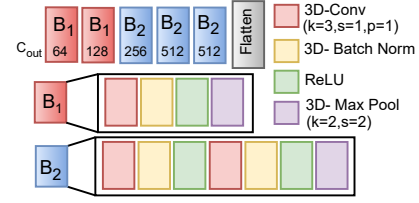


Fig. 3: Architecture of the 3D-CNN ($C3D$) for extracting the initial feature embeddings from RDI sequence. The 3D-CNN consists of two blocks B_1 and B_2 each with a combination of layers as shown.

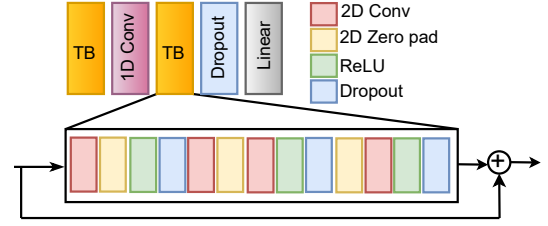


Fig. 4: Architecture of the temporal convolutional network (TCN).

extract relevant spatial and temporal features from the RDI frames. The embedding extractor consists of three sub-networks: $C3D$, TCN , and CSA . Let $E_{G_n S_k}$ represent the embedding corresponding to $G_n S_k$ and let E_Q be the embedding for a query sample. The prototypical framework determines the best representation for a class known as the prototype. Class prototype (P_{G_i}) is computed as the mean of K embeddings from the i^{th} gesture class. The query embedding E_Q is mapped to a gesture class corresponding to the closest class prototype.

1) $C3D$: The RDI frames are first passed through a 3D-CNN ($C3D$) that simultaneously extracts spatial and temporal features. $C3D$ consists of two different types of blocks as shown in Fig. 3. Block 1 (B_1) contains one 3D convolution layer, while Block 2 (B_2) contains two 3D convolution layers, each followed by batch normalisation layer, ReLU activation, and max pooling. The extracted embedding is spatially flattened before passing to the next sub-network.

2) TCN : The embeddings extracted from the RDI sequence by $C3D$ are passed to TCN to enhance its temporal content. TCN comprises of two temporal blocks (TB) stacked sequentially with dilation factors (d) of 1 and 2, respectively, as shown in Fig. 4. Each temporal block has a residual architecture. The temporal blocks consist of dilated causal 2D convolutions with a kernel size of 3 along the temporal dimension. Stacking temporal blocks with increasing d further increases the receptive field, giving the model a longer memory. Furthermore, 2D zero padding along the temporal dimension (P_T) is applied to enable causality for real-time applications and to prevent shortcut learning, forcing the model to rely on the gesture's natural temporal progression. We have also included dropout layers in TCN architecture for improved generalization and minimizing over-fitting.

3) CSA : The extracted embeddings from the $C3D$ and TCN blocks are sequentially passed through the CSA block for feature selection. The CSA block consists of two attention blocks as shown in Fig. 5a, with the detailed architecture of each attention block shown in Fig. 5b. The channel attention block (CAB) dynamically emphasizes channels with high motion saliency. The spatial attention block (SAB) emphasizes learned features of the range-Doppler cells where the motion is concentrated, thereby reducing the impact of

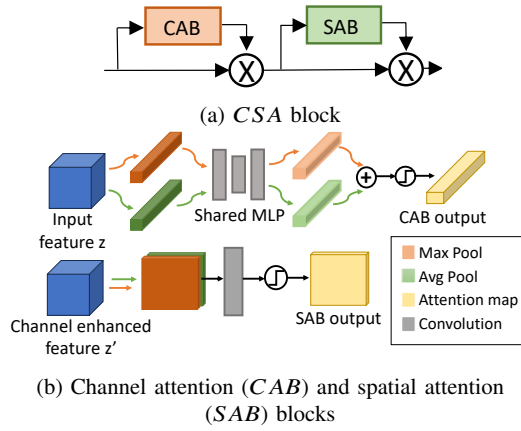


Fig. 5: (a) CSA block of the feature embedding network, (b) Architecture of channel and spatial attention blocks.

range migration and noise. This dual-attention mechanism ensures that the prototypical network learns from the most physically informative signal components rather than background clutter. The TCN output is reshaped to form the intermediate feature map $z \in \mathbb{R}^{B \times T \times HW}$ (where $HW = H \times W$). The channel attention map $M_{ch} \in \mathbb{R}^{B \times 1 \times 1}$ and spatial attention map $M_{sp} \in \mathbb{R}^{1 \times T \times HW}$ when applied to z will result in a refined feature z'' as

$$z' = M_{ch}(z) \odot z \text{ and } z'' = M_{sp}(z') \odot z' \quad (1)$$

where $\odot$ denotes the Hadamard product, B is the number of channels in the feature map, T is the temporal dimension, and HW represents the flattened spatial dimensions. For the CAB, both average and max pooling operations are performed on the extracted embedding along T and HW dimensions to obtain z_{avg}^{ch} and z_{max}^{ch} . Here, max pooling is performed to enhance the prominent features, while average pooling averages out the features in the region. These are then passed through a shared multi-layer perceptron (MLP) with one hidden layer of size $h = 100$, to obtain the attention maps. Channel attention output can thus be defined as:

$$M_{ch}(z) = \sigma(MLP(AvgPool(z)) + MLP(MaxPool(z))) \quad (2)$$

$$= \sigma(W_1(W_0(z_{avg}^{ch})) + W_1(W_0(z_{max}^{ch}))) \quad (3)$$

where $\sigma(\cdot)$ denotes the sigmoid activation function, and $W_0 \in \mathbb{R}^{h \times B}$ and $W_1 \in \mathbb{R}^{B \times h}$ denote weights of the MLP. Similarly, poolings are performed along the channel dimension (B) to obtain z_{avg}^{sp} and z_{max}^{sp} for spatial attention, as shown in Fig. 5b, resulting in the SAB output:

$$M_{sp}(z') = \sigma(f([AvgPool(z'); MaxPool(z')])) \quad (4)$$

$$= \sigma(f([z_{avg}^{sp}; z_{max}^{sp}])) \quad (5)$$

where $f(\cdot)$ denotes a convolution layer.

The proposed system design introduces a fundamental shift from existing architectures such as ML-HGR-Net [14] and attention-based relation networks [6], [7], by focusing on generating richer, physically-relevant feature embeddings. While [14] relies on multi-modal fusion, our approach achieves high discriminative power using only the RDI modality through a hierarchical feature extraction pipeline, that captures filtered spatio-temporal features. Unlike [6] and [7], which employ attention at a late-stage, our model integrates attention into the backbone as a saliency filter. This emphasizes dominant channels and range migration paths before the embedding

Table 1: Datasets along with gesture classes in training and test sets.

Dataset	No. of gesture classes	Training set (D^{train})	Test set (D^{test})
Soli [19], TinyRadarNN [20]	11	Fast swipe, Push, Finger rub, Circle, Pinch pinky	Pinch index, Palm hold, Palm tilt, Pull, Slow swipe, Finger slider
MFR [21]	12	Arm to left, Close fist horizontally, Hand away, Hand down, Hand to left, Hand rotation palm down	Arm to right, Close fist perpendicularly, Hand closer, Hand to right, Hand up, Hand rotation palm up
MCD [22]	13	Lifting left arm, Sitting down, Push, Slide left, Clockwise turning, Turning around	Lifting right arm, Standing up, Pull, Slide right, Anticlockwise turning, Walking, Waving hands

is formed, resulting in robust, physically-grounded prototypes, that enable effective discrimination in data-scarce scenarios.

IV. MODEL PERFORMANCE EVALUATION

A. Implementation Details

The proposed framework has been evaluated on four publicly available radar hand gesture datasets namely, Soli [19], TinyRadarNN [20], MFR [21], and MCD [22]. MCD dataset contains range-angle images (RAI) while the other three datasets contain RDIs¹. The train-test set split of the gesture classes was performed randomly, ensuring that opposing gestures are in different sets. The gesture classes assigned to training and test sets are shown in Table 1. Similar to [14], 150 samples of each gesture in the training set (D^{train}) and 20 samples of each gesture in the test set (D^{test}) are used for the few-shot application. The experiments are carried out using episodic training in 3-way 5-shot and 3-way 1-shot setup. The 3-way 5-shot setup indicates that in each episode, the model is provided 5 samples/class for 3 gesture classes as the support set. The model randomly chooses 3 classes from the meta training set (D^{train}) and randomly chooses 5 samples from each class. The remaining 15 samples from a gesture class are used in the query set. Due to the limited number of gesture classes in the datasets, there can be overlap in gesture classes between episodes. Training is carried out for 10,000 episodes using the Adam optimizer with a learning rate of 0.001, and testing for 2000 episodes. The sampling during testing is identical to that during training with the samples being drawn from the test set (D^{test}). There is no overlap between the classes used for meta-training and meta-testing while there is overlap in the subjects. After meta-testing, the accuracy averaged over all episodes is reported here. Random seed was fixed and identical class splits were used across all experimental runs. The model was trained and evaluated on an NVIDIA Tesla P100 GPU.

B. Results

Table 2 shows mean accuracy of the proposed C3D-CSA-TCN model along with 95% confidence interval (CI), where CI is computed using the Student's t-distribution with each episode treated as an independent and identically distributed (i.i.d.) unit. An ablation study with different combinations of sub-networks has also been performed to understand the contribution of each block to the overall model (refer Table 2). It is seen that the C3D-CSA-TCN model achieves an improvement of 2%-8% over a standard few-shot (FS) model with a ResNet-12-based feature extractor on all four datasets. The proposed model also outperforms the CSA-TCN-FS model which uses a ResNet-12 backbone instead of the C3D. The performance improvement achieved by the proposed model is least on the MCD dataset. Also, the C3D-TCN-FS model performs the best on MCD dataset which is possibly due to the MCD dataset consisting of RAI frames whereas the other three datasets consist of RDI frames. In addition, classes in the MCD dataset consist of complex actions. The

¹For further details about these datasets, please refer [19]–[22]

Table 2: Classification accuracy of the proposed C3D-CSA-TCN model and its variants (with 95% CI) on 4 datasets.

Model	Soli	TinyRadarNN	MCD	MFR	No. of Parameters
Standard FS	91.57 $\pm$ 0.48	71.20 $\pm$ 0.58	88.56 $\pm$ 0.33	82.53 $\pm$ 0.36	9,433,920
C3D-TCN-FS	93.20 $\pm$ 0.47	77.73 $\pm$ 0.64	91.21 $\pm$ 0.24	82.13 $\pm$ 0.45	27,669,913
C3D-CSA-FS	95.27 $\pm$ 0.27	77.83 $\pm$ 0.59	90.41 $\pm$ 0.59	87.96 $\pm$ 0.30	27,716,312
CSA-TCN-FS	93.69 $\pm$ 0.39	63.01 $\pm$ 0.55	89.98 $\pm$ 0.27	78.32 $\pm$ 0.38	9,488,561
C3D-CSA-TCN	97.63 $\pm$ 0.14	79.24 $\pm$ 0.52	90.73 $\pm$ 0.28	88.29 $\pm$ 0.29	27,720,433

Table 3: Classification accuracy (with 95% CI) and training and inference times of the proposed C3D-CSA-TCN model.

Dataset	3-way 5-shot			3-way 1-shot		
	Classification accuracy	Training time (s)	Inference time (s)	Classification accuracy	Training time (s)	Inference time (s)
Soli [19]	97.63 $\pm$ 0.14	4304	229	93.98 $\pm$ 0.46	2927	225
TinyRadarNN [20]	79.24 $\pm$ 0.52	2015	135	69.36 $\pm$ 0.67	1383	95
MCD [21]	90.73 $\pm$ 0.28	4208	42	76.41 $\pm$ 0.58	2308	41
MFR [22]	88.29 $\pm$ 0.29	1340	91	69.96 $\pm$ 0.55	1097	74

high parameter count of the proposed model in comparison to the standard FS model (refer Table 2) is due to the C3D backbone that significantly improves the performance.

In [19], classification accuracy of 92.10% was achieved on the Soli dataset, when trained and tested on 1000 samples each, using a random forest classifier. However, when the model in [19] is trained with only 5 samples each from 3 gesture classes and tested on 1000 samples, it achieves an accuracy of only 52.24%. Using the same data in a few-shot framework achieves much better performance.

Table 3 compares the performance of the C3D-CSA-TCN model under 5-shot and 1-shot settings. It is seen that the accuracy drops by 4-14% across the four datasets when the number of samples per class is reduced from 5 to 1. Since 1-shot uses only one sample per class in the support set, as expected it results in lower performance. However, it is to be noted that the model is able to correctly classify gestures with reasonable accuracy using only one sample per class.

Cross-dataset generalization was not evaluated due to varying radar data formats. Retraining/fine-tuning is expected when transitioning between different radar types to account for varying signal characteristics. In Table 4, we evaluate the performance of the C3D-CSA-TCN model with different loss functions. We consider cross-entropy (CE) loss function, a combination of CE loss and triplet loss, natural log likelihood (NLL) loss, and contrastive loss [23]. The CE loss tries to minimise the distance between the model prediction and ground truth. The triplet loss creates positive and negative anchors and tries to reduce the relative distance of the anchor to the positive anchor with respect to the negative anchor. The NLL loss involves a softmax layer followed by CE loss. The contrastive loss considers an input pair of samples that are either similar or dissimilar, and brings the similar ones closer and dissimilar ones farther. From Table 4, we see that the proposed model using CE loss provides consistently strong results across datasets. However, on the Soli dataset the proposed model performs the best using NLL loss. Adding triplet loss to the CE loss does not provide any additional advantage. In contrast, the model performs poorly with contrastive loss as it enforces an absolute minimum separation between classes. In few-shot episodes with few samples and similar looking radar data, this separation distorts the feature space and hinders learning of effective prototypes.

V. CONCLUSION

We have introduced a novel few-shot architecture for radar-based hand gesture recognition. The proposed framework uses a novel feature embedding network with three sub-networks, namely, C3D,

Table 4: Classification accuracy of the proposed C3D-CSA-TCN model with different loss functions.

Dataset	CE	CE + Triplet	NLL	Contrastive
Soli [19]	97.63 $\pm$ 0.14	94.94 $\pm$ 0.43	98.90 $\pm$ 0.12	6.93 $\pm$ 0.66
TinyRadarNN [20]	79.24 $\pm$ 0.52	78.16 $\pm$ 0.45	72.84 $\pm$ 0.45	13.63 $\pm$ 1.15
MFR [22]	88.29 $\pm$ 0.31	87.21 $\pm$ 0.31	86.58 $\pm$ 0.31	6.66 $\pm$ 0.63

TCN, and CSA that enable extraction of spatial and temporal features from radar imagery. The model is able to classify gestures with very good accuracy using even only one sample per class. The model performance was validated on four datasets. The contribution of each sub-block in the feature embedding network was determined through an ablation study. It was observed that the C3D-CSA-TCN model with CE loss achieves the best performance, validating the effectiveness of the proposed architecture.

REFERENCES

- [1] Y. Wand *et al.*, “Generalizing from a few examples: A survey on few-shot learning,” *ACM Comput. Surv.*, vol. 53, no. 3, pp. 63:1–63:34, 2020.
- [2] G. Koch, R. Zemel, and R. Salakhutdinov, “Siamese neural networks for one-shot image recognition,” in *Proc. Int. Conf. Mach. Learn. (ICML) Deep Learn. Workshop*, vol. 2, 2015.
- [3] O. Vinyals *et al.*, “Matching networks for one-shot learning,” *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 29, 2016.
- [4] F. Sung *et al.*, “Learning to compare: Relation network for few-shot learning,” in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognition*, pp. 1199–1208, 2018.
- [5] J. Snell, K. Swersky, and R. Zemel, “Prototypical networks for few-shot learning,” *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 30, 2017.
- [6] B. Hui *et al.*, “Self-attention relation network for few-shot learning,” in *Proc. IEEE Int. Conf. Multimedia & Expo Workshops (ICMEW)*, pp. 198–203, 2019.
- [7] Z. Li *et al.*, “SABERNet: Self-attention based effective relation network for few-shot learning,” *Pattern Recognit.*, vol. 133, p. 109024, 2023.
- [8] E. Rahimian *et al.*, “FS-HGR: Few-shot learning for hand gesture recognition via electromyography,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 1004–1015, 2021.
- [9] N. Kimura, “Self-supervised approach for few-shot hand gesture recognition,” in *Proc. ACM Symp. User Interface Software and Technol.*, pp. 1–4, 2022.
- [10] C. Ma *et al.*, “Skeleton-based dynamic hand gesture recognition using an enhanced network with one-shot learning,” *Appl. Sciences*, vol. 10, no. 11, p. 3680, 2020.
- [11] Z. Liu, C. Pan, and H. Wang, “Continuous gesture sequences recognition based on few-shot learning,” *Int. J. Aerosp. Eng.*, vol. 2022, 2022.
- [12] X. Zeng, C. Wu, and W.-B. Ye, “User-definable dynamic hand gesture recognition based on doppler radar and few-shot learning,” *IEEE Sensors J.*, vol. 21, no. 20, pp. 23224–23233, 2021.
- [13] G. Mauro *et al.*, “Few-shot user-definable Radar-based hand gesture recognition at the edge,” *IEEE Access*, vol. 10, pp. 29741–29759, 2022.
- [14] X. Shen *et al.*, “ML-HGR-Net: A meta-learning network for FMCW radar-based hand gesture recognition,” *IEEE Sensors J.*, vol. 22, no. 11, pp. 10808–10817, 2022.
- [15] S. Woo *et al.*, “CBAM: Convolutional block attention module,” in *Proc. European Conf. Comput. Vis. (ECCV)*, pp. 3–19, 2018.
- [16] B. Min *et al.*, “Data augmentation method for plant leaf disease recognition,” *Appl. Sciences*, vol. 13, no. 3, p. 1465, 2023.
- [17] P. Li *et al.*, “Multi-scale residual denoising GAN model for producing super-resolution CTA images,” *J. Ambient. Intell. Humaniz. Comput.*, pp. 1–10, 2022.
- [18] C. Du *et al.*, “Enhanced multi-channel feature synthesis for hand gesture recognition based on CNN with a channel and spatial attention mechanism,” *IEEE Access*, vol. 8, pp. 144610–144620, 2020.
- [19] J. Lien *et al.*, “Soli: Ubiquitous gesture sensing with millimeter wave radar,” *ACM Trans. Graph.*, vol. 35, no. 4, pp. 1–19, 2016.
- [20] M. Scherer *et al.*, “TinyRadarNN: Combining spatial and temporal convolutional neural networks for embedded gesture recognition with short range radars,” *IEEE Internet Things J.*, vol. 8, no. 13, pp. 10336–10346, 2021.
- [21] Y. Li *et al.*, “Towards domain-independent and real-time gesture recognition using mmWave signal,” *IEEE Trans. Mobile Comput.*, 2022.
- [22] P. Grobelny and A. Narbudowicz, “Hand gestures recorded with mm-Wave FMCW radar (AWR1642),” 2021. IEEE Dataport, DOI: 10.21227/wh5w-c362.
- [23] J. Terven *et al.*, “A comprehensive survey of loss functions and metrics in deep learning,” *Artif. Intell. Rev.*, vol. 58, no. 7, p. 195, 2025.