FAMNet: Frequency-Aware Adaptive Multi-Scale Network for Multivariate Time Series Classification

Liyu Luo, Jiyao Gao, Jiaxuan Xu, Qingbao Guan, Jie Zuo*, Lei Duan
School of Computer Science, Sichuan University, Chengdu, China
luoliyu@stu.scu.edu.cn, jiyao.gao@outlook.com, {xujx, guanqingbao}@stu.scu.edu.cn, {zuojie, leiduan}@scu.edu.cn

***Abstract*—In time series analysis, Multivariate Time Series Classification (MTSC) plays a crucial role in diverse real-world applications. Recently, integrating multi-scale analysis into deep learning models has been effective in improving performance for MTSC. However, existing multi-scale methods typically employ fixed scales, which prevents models from selecting the most discriminative scales for each input. To address this limitation, we propose the Frequency-aware Adaptive Multi-scale Network (FAMNet) for MTSC. Our method uses the Fourier transform to extract frequency characteristics of the input time series and scores candidate scales accordingly. To make scale selection trainable, we introduce Gumbel-Softmax, allowing the model to adaptively select multiple suitable scales per sample. The selected scales are then used for multi-scale patch embedding and integrated through cross-scale fusion for hierarchical representation. Experiments on 20 datasets demonstrate that the proposed FAMNet achieves the best accuracy compared to advanced baselines in MTSC tasks.**

***Index Terms*—multivariate time series classification, frequency-aware adaptive scale selection, hierarchical cross-scale fusion**

I. Introduction

Multivariate Time Series (MTS) refers to a collection of data points formed by temporally sampling correlated signals, where each time step contains values of multiple variables. Multivariate Time Series Classification (MTSC) is a fundamental task in MTS analysis, which learns discriminative features from MTS to achieve accurate classification. Owing to its wide application in human activity recognition [1], healthcare monitoring [2], and industrial fault diagnosis [3], MTSC has received significant attention.

In recent years, deep learning models have been widely applied to MTSC [4], [5], [6]. Since time series often exhibit distinct patterns at different temporal scales, multi-scale methods have achieved success due to their ability to analyze hierarchical information by extracting features at different scales [7]. Based on how they extract multi-scale features, these methods can be grouped into three categories, which are illustrated in Fig. 1. The first is multi-level methods [8], which feed the time series through multiple levels of the same module. Each level uses different preset hyperparameters so that the series is analyzed at different scales. The second is downsampling-based methods [9], which repeatedly downsample the time series (e.g., via moving average). As the number of downsampling stages increases, the model can extract temporal features from fine-grained to coarse-grained. The third is multi-scale patch-based methods [10]. Motivated by the ability of the patch technique to capture local temporal patterns [11], these models use different kernel sizes or patch lengths to obtain patches at multiple scales, so that local features are captured at different scales. Together, these methods have been shown to be effective for MTSC.

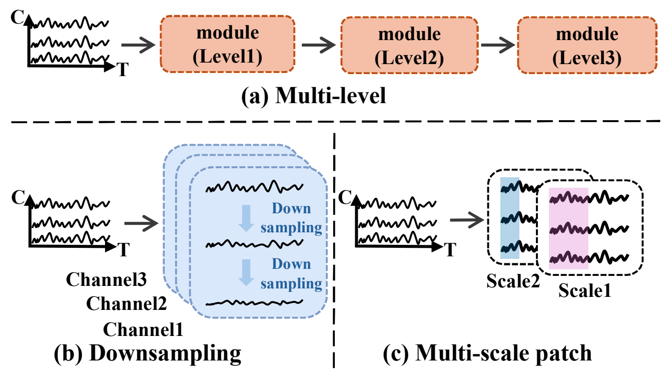

Fig. 1. Three categories of multi-scale methods.

However, existing multi-scale methods typically rely on fixed scale setting, which are determined before training and applied uniformly to all inputs. In multi-level methods, the hyperparameters of each level are preset and fixed; in downsampling-based methods, the sampling ratios are fixed; in methods using multi-scale patches, the set of kernel sizes or patch lengths is fixed. Such a design leads to two main limitations. First, the fixed scale setting lacks generality, because scales that are optimal for one sample are not necessarily optimal for another. For example, in human activity recognition, scales that capture short-term gestures such as hand movements often differ from those suited to long-duration activities such as walking or standing, so the fixed scale setting may not suit both types of patterns. Second, not all predefined scales have a positive effect on classification. Redundant scales can introduce noise and hinder the learning of discriminative representations [12], thus complicating analysis for MTS.

To address these limitations, we propose the Frequency-aware Adaptive Multi-scale Network (FAMNet), a multi-scale MTSC framework that adaptively selects scales for each input based on its frequency characteristics. We use the Fourier transform to obtain a frequency representation of each input,

*Corresponding author: Jie Zuo (email: zuojie@scu.edu.cn)

because it can provide multi-scale frequency and periodic information from the time series itself without any preset. However, the resulting periodic components are scattered and irregular across samples, and using them directly for multi-scale analysis would cause dimension misalignment and unstable optimization during training. We therefore introduce Gumbel-Softmax [13] to learn the discrete frequency-to-scale mapping in a differentiable manner. The above steps form the Frequency-aware Adaptive Scale Selection (FASS) module. FASS addresses the fixed scale setting problem through adaptive scale selection per input. By selecting scales based on each input's frequency, it also focuses on informative scales and reduces the impact of redundant ones. The selected scales are then used to construct multi-scale patch embeddings. These embeddings are integrated through the Hierarchical Cross-scale Fusion (HCF) module. Within the HCF, intra-scale attention first captures local patterns at each scale independently. These local patterns are then aggregated across scales by inter-scale attention to obtain global context, which is distributed back to each scale via a gating mechanism so that each scale's local patterns are enhanced with global context. In this way, FAMNet combines frequency-aware adaptive scale selection with cross-scale fusion, effectively analyzing time series.

In summary, our main contributions are as follows:

- We propose FAMNet, a multi-scale MTSC framework that adaptively selects scales via frequency-aware scale selection and fuses multi-scale representations through cross-scale fusion.
- We achieve adaptive scale selection by exploiting the frequency domain and the Gumbel-Softmax technique, enabling differentiable learning of scale selection.
- We conduct extensive experiments on 20 datasets and demonstrate that FAMNet achieves the best accuracy compared to advanced baselines in MTSC tasks.

II. Related Work

A. Multivariate Time Series Classification

MTSC has been extensively studied in recent years. Early approaches relied on traditional machine learning methods such as distance-based classifiers [14] and feature-based methods [15]. With the development of deep learning, various neural network architectures have been applied to MTSC. For example, CNN-based models have shown remarkable success in capturing local temporal patterns through temporal convolutions [16]. RNN-based models, including the LSTM and GRU variants, have been used to model sequential dependencies in time series data [17]. More recently, Transformer-based models have gained significant attention in MTSC by effectively modeling temporal relationships across time steps [5], [18]. In addition, other methods such as GNN-based models [19] have been proposed for MTSC, further enriching the range of deep learning solutions for this task.

B. Multi-Scale Analysis in Time Series

Multi-scale analysis has proven to be effective for time series analysis, as time series often exhibit multiple temporal patterns at different scales. FormerTime [8] employs hierarchical Transformer blocks with different hyperparameters to generate multi-scale feature maps, capturing patterns at various temporal resolutions. TimeMixer++ [9] achieves multi-scale analysis through downsampling strategies to extract temporal patterns at different resolutions. MedSpaformer [10] utilizes patch-based multi-granularity encoding schemes to capture intra- and inter-granularity temporal dependencies on different scales. These approaches focus on different strategies for extracting multi-scale representations. Besides, some recent studies have shifted attention to improving the effectiveness of multi-scale feature utilization. For example, DisMS-TS [12] first introduces a temporal disentanglement module to eliminate redundant scale-shared features, ensuring the consistency of scale-shared representations and the disparity of scale-specific representations across different temporal scales.

However, a key limitation in many of these methods is that fixed and predefined scales fail to adapt to the diverse characteristics of different time series. This limitation often leads to suboptimal scale selection, as irrelevant scales may introduce noise and interfere with discriminative pattern learning. Moreover, manually adjusting scale combinations across datasets increases training complexity and limits generalization. Our method addresses this limitation by enabling adaptive scale selection for each input time series, allowing the model to dynamically choose the most suitable scales.

C. Frequency Domain Analysis in Time Series

Frequency domain analysis has been widely used in signal processing and time series analysis. Some recent works have incorporated frequency domain information into deep learning models for time series tasks. FreTS [20] operates MLPs in the frequency domain through complex frequency representations. MF-CLR [21] uses hierarchical contrastive learning across frequencies with cross-frequency consistency. FreRA [22] augments time series in the frequency domain by separating critical and unimportant frequency components. These methods typically extract frequency components using Fourier transform or wavelet transform and combine them with time-domain features through concatenation or hybrid architectures. However, they primarily treat frequency information as supplementary features without fully exploiting its potential for adaptive modeling. Motivated by the ability of frequency domain analysis to extract periodic patterns from time series, our work leverages frequency information combined with Gumbel-Softmax to dynamically select appropriate scales for each input time series.

III. Methodology

A. Problem Formulation

The task of MTSC is to learn a mapping function from a given multivariate time series to a corresponding categorical label. Formally, let a dataset be denoted as $\mathcal{D} = (X_i, y_i)_{i=1}^{M}$, where M is the number of samples. Each sample consists of a multivariate time series $X_i \in \mathbb{R}^{L \times C}$ and its associated class label $y_i \in \{1, 2, ..., K\}$, where L is the sequence length, C is

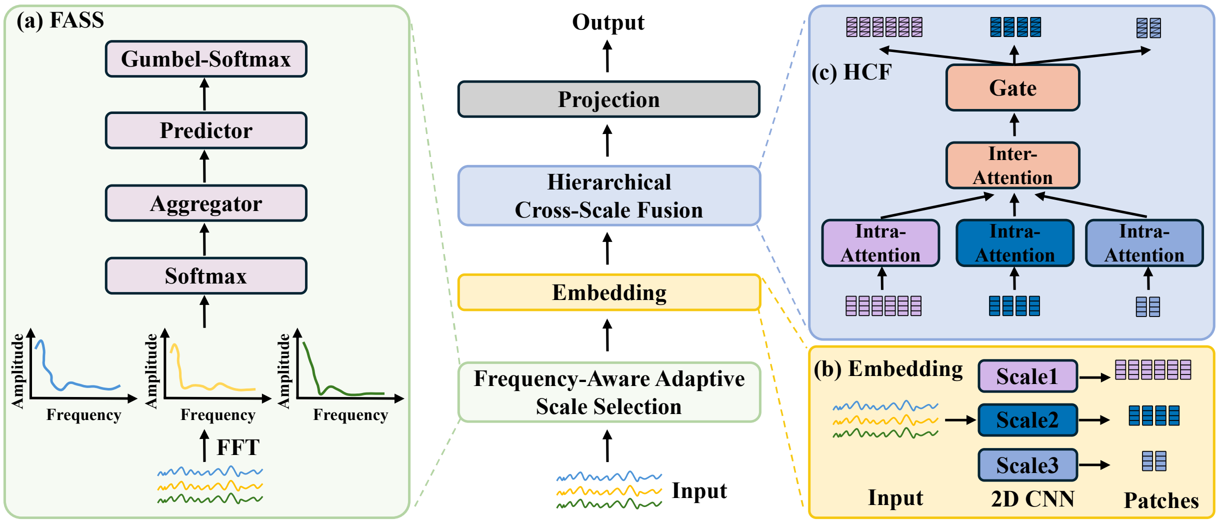

Fig. 2. Overview of the Frequency-aware Adaptive Multi-scale Network (FAMNet).

the number of variables, and K is the total number of classes. The primary objective of MTSC is to learn a model f_θ with parameters θ that can accurately predict the label for an unseen time series, i.e., $\hat{y} = f_\theta(X)$.

B. Overall Framework

Fig. 2 illustrates the overall architecture of our proposed FAMNet, which is designed to adaptively select optimal scales and fuse multi-scale information for MTSC. For a given MTS input, the FASS module first analyzes the frequency characteristics of the input through channel-aware frequency analysis and frequency-discriminative learning. Based on the learned frequency representations, FASS employs Gumbel-Softmax to dynamically select k most suitable scales from a predefined scale pool. The selected scales are then used to generate multi-scale patch embeddings through dynamic patch embedding layers. Subsequently, the HCF module processes these embeddings through intra-scale attention to capture temporal dependencies within each scale, followed by inter-scale attention to enable information exchange across different scales. The cross-scale information is then fused back to each scale using a gate mechanism, allowing the model to enhance local patterns with global multi-scale insights. Finally, the processed multi-scale representations are concatenated and fed into a classification head to produce the final predictions.

C. Frequency-Aware Adaptive Scale Selection Module

The FASS module adaptively selects the most discriminative patch scales for each input sample by analyzing its frequency characteristics.

Given a MTS input $\mathbf{X} \in \mathbb{R}^{L\times C}$, we first transform each channel to the frequency domain using the Fast Fourier Transform (FFT) to obtain the magnitude spectrum $\mathbf{M}_c$ for each channel c:

$$\mathbf{S}_c = \mathrm{FFT}(\mathbf{X}_c), \quad \mathbf{M}_c = |\mathbf{S}_c|, \tag{1}$$

where $\mathbf{S}_c$ and $\mathbf{M}_c$ represent the complex spectrum and magnitude spectrum of channel c. To aggregate multi-channel frequency information, we employ a channel attention mechanism that learns the importance of each channel:

$$\mathbf{f}_c = \mathrm{Mean}(\mathbf{M}_c), \quad \boldsymbol{\alpha} = \mathrm{Softmax}(\mathrm{MLP}(\mathbf{f})), \tag{2}$$

where $\mathbf{f}_c$ is the global frequency feature of channel c, and $\boldsymbol{\alpha} \in \mathbb{R}^C$ denotes the learned channel attention weights. The aggregated spectrum is then computed as:

$$\mathbf{M}_{\text{agg}} = \sum_{c=1}^{C} \alpha_c \cdot \mathbf{M}_c. \tag{3}$$

To map frequency characteristics to scale preferences, we encode the aggregated spectrum into a discriminative representation and predict scale logits:

$$\mathbf{h} = \mathrm{Encoder}(\mathrm{Normalize}(\mathbf{M}_{\text{agg}})), \quad \mathbf{s} = \mathrm{Predictor}(\mathbf{h}), \tag{4}$$

where Encoder is a two-layer MLP with LayerNorm and ReLU activation, and Predictor outputs logits for each candidate scale in the predefined scale pool $\mathcal{S} = \{s_1, s_2, \ldots, s_N\}$.

To enable end-to-end training while selecting discrete scales, we employ the Gumbel-Softmax trick to iteratively select k non-repeating scales:

$$\begin{aligned} \mathbf{p}^{(i)} &= \text{Gumbel-Softmax}(\mathbf{s}^{(i)}, \tau), \\ \mathbf{s}^{(i+1)} &= \mathbf{s}^{(i)} - \mathbf{p}^{(i)} \cdot M, \end{aligned} \tag{5}$$

where τ is the temperature parameter, M is a large masking value, and $\mathbf{p}^{(i)}$ is the selection vector in iteration i. During training, we set $\tau = 1.0$ for exploration, while during inference, we use $\arg\max$ for deterministic selection.

Since different samples in a batch may select different scales, we employ a batch-level voting mechanism to determine a unified set of scales for efficient processing. We collect all selected scale indices across the batch, count their frequencies, and select the top-k most frequent non-repeating scales as $\mathcal{S}_{\text{selected}} = \{s_1, s_2, \ldots, s_k\}$.

Based on the selected scales, we dynamically generate multi-scale patch embeddings using convolutional layers:

$$\mathbf{E}_i = \text{Conv2D}(\mathbf{X}, \text{kernel\_size} = (C, s_i)) + \text{PE}, \tag{6}$$

where PE denotes positional embeddings.

D. Hierarchical Cross-Scale Fusion Module

The HCF module enables information interaction and fusion across different scales through a hierarchical attention mechanism.

For each scale $i \in \{1, 2, \ldots, k\}$, we apply intra-scale attention to capture temporal dependencies within that scale. Given an input list of patch embeddings $\{\mathbf{E}_i\}_{i=1}^{k}$, where $\mathbf{E}_i$ represents the embedding for scale i and k is the number of selected scales, the intra-scale attention is formulated as:

$$\mathbf{H}_i = \text{Attn}_{\text{intra}}(\mathbf{E}^{(i)}, \mathbf{E}^{(i)}, \mathbf{E}^{(i)}). \tag{7}$$

Attention blocks for different scales do not share parameters, ensuring that each block learns distinct patterns. Note that since the selected scale combinations may vary during training, the attention blocks learn position-based patterns rather than features specific to particular scale values, enabling better generalization across different scale combinations.

To enable information exchange across scales, we obtain scale-level representations by global average pooling and then apply inter-scale attention:

$$\mathbf{R}_i = \text{AvgPool}(\mathbf{H}_i), \quad \mathbf{R}_{\text{fused}} = \text{Attn}_{\text{inter}}(\mathbf{R}, \mathbf{R}, \mathbf{R}), \tag{8}$$

where $\mathbf{R} = [\mathbf{R}_1, \mathbf{R}_2, \ldots, \mathbf{R}_k]$ is the concatenation of all scale-level representations, and $\mathbf{R}_{\text{fused}}$ represents the fused cross-scale information.

We fuse the cross-scale information back to each scale using a gate mechanism, which controls the information flow according to relevance:

$$\begin{aligned} \mathbf{g}_i &= \sigma(\text{Mean}(\mathbf{H}_i \odot \text{Broadcast}(\mathbf{R}_{\text{fused}}))), \\ \mathbf{H}_i^{\text{final}} &= \mathbf{H}_i + \mathbf{g}_i \odot \text{Broadcast}(\mathbf{R}_{\text{fused}}), \end{aligned} \tag{9}$$

where $\sigma(\cdot)$ denotes the Sigmoid function, $\odot$ denotes element-wise multiplication, and Broadcast expands the scale-level representation to match the sequence length of $\mathbf{H}_i$.

Finally, we concatenate all scale representations and feed the result into a classification head:

$$\begin{aligned} \mathbf{h}_{\text{global}} &= \text{GlobalAvgPool}([\mathbf{H}_1^{\text{final}}, \mathbf{H}_2^{\text{final}}, \ldots, \mathbf{H}_k^{\text{final}}]), \\ \mathbf{y} &= \text{MLP}(\text{LayerNorm}(\mathbf{h}_{\text{global}})). \end{aligned} \tag{10}$$

E. Time Complexity Analysis

Let L be the sequence length, C the number of variables, k the number of selected scales, and d the hidden dimension. The FASS module costs $\mathcal{O}(CL\log L)$ for FFT and $\mathcal{O}(CLd)$ for the scoring network. Gumbel-Softmax over the scale pool adds negligible cost. Patch embedding for the k selected scales costs $\mathcal{O}(kLd)$. The HCF module is dominated by intra-scale attention, costing $\mathcal{O}(L^2d)$. Inter-scale attention then fuses information across the k scale-level vectors at much lower cost. Overall, the main complexity is $\mathcal{O}(CL\log L + L^2d)$, on the same order as standard Transformer-based models with sequence length L, with an extra $\mathcal{O}(CL\log L)$ from frequency-aware scale selection.

IV. Experiments

A. Experimental Setup

1) Datasets: Based on common practice in MTSC evaluation and the availability of reported baseline results, we conduct experiments on 20 datasets from the UEA multivariate time series classification archive [23], a widely used benchmark for MTSC. These datasets cover diverse real-world scenarios and exhibit significant variations in time series length, dimensionality, and other characteristics.

2) Baselines: We compare our method FAMNet with eight representative baselines, including two CNN-based models: OS-CNN and MOS-CNN [16]; two Transformer-based models: SVP-T [18] and ConvTran [5]; two multi-scale models: FormerTime [8] and MedFormer [2]; one GNN-based model: TodyNet [19]; and one FFT-based model: MPTSNet [24].

3) Implementation Details: Experiments are conducted on Ubuntu 22.04 using a single NVIDIA RTX 4090D GPU, with PyTorch 2.5.1 and Python 3.12. We train all models using AdamW optimizer to minimize the cross-entropy loss, with an initial learning rate of 0.001, weight decay of 0.01, batch size of 16, and train for 100 epochs. Following common practice in multi-scale time series analysis studies, we set the candidate scale pool as $\{2, 4, 8, 16, 32, 48, 64\}$. The performance is evaluated by computing the accuracy on each dataset, the average accuracy and average rank across datasets, and the number of datasets on which each method achieves the top-1 accuracy.

B. Experimental Results

Table I summarizes the classification accuracy of FAMNet and eight comparative methods on 20 UEA datasets. Baseline accuracy is taken from the respective papers. For FormerTime and MedFormer, we reproduce their results by running their implementations under our environment, as their original work does not report results on all datasets used here. All results are retained to three decimal places, with the best result per row in bold and the second best underlined.

As Table I shows, FAMNet achieves the best overall performance among all methods. It obtains the lowest average rank (2.550) and the highest average accuracy (0.807), and achieves the top-1 accuracy on 10 out of 20 datasets. Compared with MPTSNet, the runner-up in average accuracy (0.786), FAMNet achieves better performance on all these metrics. On specific datasets, FAMNet achieves the best accuracy on EthanolConcentration and StandWalkJump, which are known for complex temporal patterns, and also ranks first on DuckDuckGeese, ERing and so on. On datasets where FAMNet does not rank first, it still places second or third in most cases, indicating stable generalization across domains such as human activity recognition, healthcare, and motion recognition. These results validate the effectiveness of FAMNet for MTSC.

TABLE I
ACCURACY OF PROPOSED FAMNET AND 8 COMPARATIVE METHODS ON 20 DATASETS

Datasets	OS-CNN	MOS-CNN	SVP-T	ConvTran	FormerTime	MedFormer	TodyNet	MPTSNet	**FAMNet**
ArticularyWordRecognition	0.988	0.991	**0.993**	0.983	0.983	0.987	0.987	0.977	0.990
AtrialFibrillation	0.233	0.183	0.400	0.400	**0.600**	0.467	0.467	0.533	0.533
BasicMotions	**1.000**	**1.000**	**1.000**	**1.000**	0.925	**1.000**	**1.000**	**1.000**	**1.000**
Cricket	0.993	0.990	**1.000**	**1.000**	0.986	0.972	**1.000**	0.944	**1.000**
DuckDuckGeese	0.540	0.615	0.700	0.620	0.600	0.580	0.580	0.680	**0.740**
Epilepsy	0.980	**0.996**	0.986	0.986	0.935	0.971	0.971	0.971	0.986
ERing	0.881	0.915	0.937	0.963	0.933	0.941	0.915	0.944	**0.978**
EthanolConcentration	0.240	0.415	0.331	0.361	0.426	0.346	0.350	0.433	**0.441**
FingerMovements	0.568	0.568	0.600	0.560	0.620	0.580	0.570	0.640	**0.660**
Heartbeat	0.489	0.604	**0.790**	0.785	0.785	0.741	0.756	0.756	0.771
Libras	0.950	**0.965**	0.883	0.928	0.850	0.889	0.850	0.872	0.894
LSST	0.413	0.521	0.666	0.616	0.517	0.634	0.615	0.604	**0.686**
MotorImagery	0.535	0.515	0.670	0.560	**0.690**	0.580	0.640	0.650	0.640
NATOPS	0.968	0.951	0.906	0.944	0.961	0.967	**0.972**	0.944	0.944
PenDigits	0.985	0.983	0.983	0.987	0.970	0.977	0.987	**0.989**	**0.989**
RacketSports	0.877	**0.929**	0.842	0.862	0.822	0.829	0.803	0.875	0.842
SelfRegulationSCP1	0.835	0.829	0.884	0.918	0.908	**0.932**	0.898	0.928	0.911
SelfRegulationSCP2	0.532	0.510	0.600	0.583	0.583	0.578	0.550	0.572	**0.611**
StandWalkJump	0.383	0.383	0.467	0.333	0.600	0.533	0.467	0.533	**0.667**
UWaveGestureLibrary	0.927	0.926	**0.941**	0.891	0.906	0.856	0.850	0.881	0.856
Average Rank	5.950	5.350	3.800	4.300	5.250	5.100	5.250	4.250	**2.550**
Average Accuracy	0.716	0.739	0.779	0.764	0.780	0.768	0.761	0.786	**0.807**
Number of Top-1	2	4	5	2	2	2	3	2	**10**

TABLE II
PERFORMANCE OF MODEL VARIANTS AND SCALE SETTINGS

Model Variants	Average Accuracy	**Scale settings**	Average Accuracy
FAMNet	**0.807**	Adaptive	**0.807**
w/o FFT	0.767	$\{2, 4, 8\}$	0.774
w/o CF	0.750	$\{2, 4, 8, 16\}$	0.786
w/o GS	0.776	$\{2, 4, 8, 16, 32\}$	0.778

C. Ablation Study

To assess the contribution of key components and the benefit of adaptive scale selection, we conduct two groups of ablation studies on 20 datasets. The first group shows four settings: the full FAMNet and three variants—without FFT, without cross-scale fusion (CF), and without Gumbel-Softmax (GS). The second group compares FAMNet under adaptive scale selection with several fixed scale settings. As Table II shows, removing any of these components reduces average accuracy, and adaptive scale selection achieves higher average accuracy than the fixed scale combinations. These results underscore the importance of the frequency-aware module, cross-scale fusion, and differentiable scale selection in FAMNet, as well as the advantage of adaptive over fixed scale selection for MTSC.

D. Hyperparameter Sensitivity Analysis

The number of selected scales k is the main hyperparameter in FAMNet, as it determines how many scales are used for multi-scale patch embedding. We analyze the sensitivity of k by varying it from 1 to 7 and report the average accuracy over all 20 datasets as well as the accuracy on two representative datasets, DuckDuckGeese and FingerMovements.

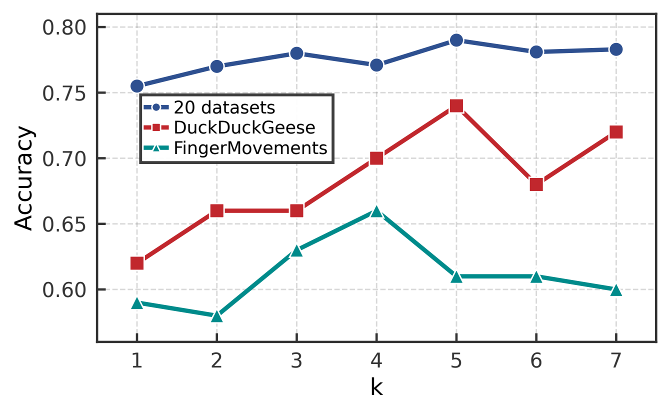

Fig. 3. Hyperparameter sensitivity analysis: accuracy as the number of selected scales k varies, including average accuracy over 20 datasets and accuracy on DuckDuckGeese and FingerMovements.

As shown in Fig. 3, the average accuracy over the 20 datasets first improves as k increases and then tends to level off or decline when k is too large. This trend indicates that using more scales helps capture multi-scale temporal patterns, but an excessively large k can introduce redundant or noisy scales and hurt performance.

The two curves for DuckDuckGeese and FingerMovements show that the best k differs across datasets: one dataset peaks at $k = 4$, the other peaks at $k = 5$. This indicates that the optimal number of scales is related to data, and a single fixed k may not suit all datasets. Together with the trend of average accuracy, these findings motivate adaptive choice k for different samples or datasets as a direction for future work.

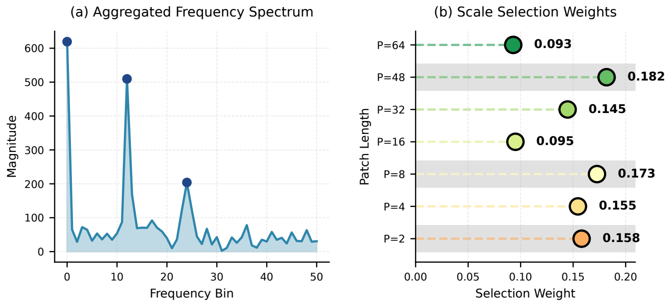

Fig. 4. Interpretability analysis: aggregated frequency spectrum and scale selection weights for a sample from BasicMotions ($k = 3$).

E. Interpretability

We provide an interpretability analysis of FAMNet. Fig. 4 shows the aggregated frequency spectrum of the input (subfigure a) and the scale selection weights before Gumbel-Softmax (subfigure b) for a representative sample randomly selected from the BasicMotions dataset. The scale weights reflect the scores assigned to each candidate scale from the frequency representation. We observe that the pattern of scale weights aligns with the frequency content: when the spectrum concentrates in certain frequency bands, the corresponding scales receive higher weights, and the model tends to select them after Gumbel-Softmax. This demonstrates that FASS adapts scale selection to the input's frequency structure and that the chosen scales are interpretable in terms of the signal's frequency content.

V. Conclusion

In this paper, we propose FAMNet, a multi-scale model for MTSC. FAMNet adaptively selects scales per sample via frequency-aware scale selection with the Fourier transform and Gumbel-Softmax, and fuses multi-scale representations through cross-scale fusion. Extensive experiments on 20 datasets demonstrate the effectiveness of FAMNet, providing a valuable and reliable solution for the MTSC tasks. In future work, we intend to extend the frequency-aware adaptive scale selection to other time series tasks and settings.

References

[1] E. Ramirez, M. Wimmer, and M. Atzmueller, "A computational framework for interpretable anomaly detection and classification of multivariate time series with application to human gait data analysis," in *International Workshop on Knowledge Representation for Health Care*. Springer, 2019, pp. 132–147.

[2] Y. Wang, N. Huang, T. Li, Y. Yan, and X. Zhang, "Medformer: A multi-granularity patching transformer for medical time-series classification," in *Advances in Neural Information Processing Systems*, vol. 37, 2024, pp. 36 314–36 341.

[3] S. Dheepadharshani, S. Anandh, K. Bhavinaya, and R. Lavanya, "Multivariate time-series classification for automated fault detection in satellite power systems," in *2019 International Conference on Communication and Signal Processing*. IEEE, 2019, pp. 0814–0817.

[4] A. Dempster, D. F. Schmidt, and G. I. Webb, "Minirocket: A very fast (almost) deterministic transform for time series classification," in *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 2021, pp. 248–257.

[5] N. M. Foumani, C. W. Tan, G. I. Webb, and M. Salehi, "Improving position encoding of transformers for multivariate time series classification," *Data mining and knowledge discovery*, vol. 38, pp. 22–48, 2024.

[6] Y. Wen, T. Ma, R. Luss, D. Bhattacharjya, A. Fokoue, and A. A. Julius, "Shedding light on time series classification using interpretability gated networks," in *The Thirteenth International Conference on Learning Representations*, 2025.

[7] F. Gao, L. Zhang, and Y. Zhang, "Fdnet: A novel multivariate time series classification model through fusing feature and difference," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 5725–5729.

[8] M. Cheng, Q. Liu, Z. Liu, Z. Li, Y. Luo, and E. Chen, "Formertime: Hierarchical multi-scale representations for multivariate time series classification," in *Proceedings of the ACM web conference*, 2023, pp. 1437–1445.

[9] S. Wang, J. Li, X. Shi, Z. Ye, B. Mo, W. Lin, S. Ju, Z. Chu, and M. Jin, "Timemixer++: A general time series pattern machine for universal predictive analysis," in *The Thirteenth International Conference on Learning Representations*, 2025.

[10] J. Ye, W. Zhang, Z. Li, J. Li, and F. Tsung, "Medspaformer: a transferable transformer with multi-granularity token sparsification for medical time series classification," 2025.

[11] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers," in *The Eleventh International Conference on Learning Representations*, 2023.

[12] Z. Liu, P. Duan, B. Wang, X. Tang, Q. Chu, C. Zhang, Y. Huang, and B. Zhang, "Disms-ts: Eliminating redundant multi-scale features for time series classification," in *Proceedings of the ACM International Conference on Multimedia (MM)*, 2025.

[13] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," in *International Conference on Learning Representations*, 2017.

[14] M. Shokoohi-Yekta, B. Hu, H. Jin, J. Wang, and E. Keogh, "Generalizing dtw to the multi-dimensional case requires an adaptive approach," *Data mining and knowledge discovery*, vol. 31, pp. 1–31, 2017.

[15] J. Lines, L. M. Davis, J. Hills, and A. Bagnall, "A shapelet transform for time series classification," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2012, pp. 289–297.

[16] W. Tang, G. Long, L. Liu, T. Zhou, M. Blumenstein, and J. Jiang, "Omni-scale CNNs: a simple and effective kernel size configuration for time series classification," in *International Conference on Learning Representations*, 2022.

[17] F. Karim, S. Majumdar, H. Darabi, and S. Harford, "Multivariate lstm-fcns for time series classification," *Neural Networks*, vol. 116, p. 237–245, Aug. 2019.

[18] R. Zuo, G. Li, B. Choi, S. S. Bhowmick, D. N.-y. Mah, and G. L. Wong, "Svp-t: A shape-level variable-position transformer for multivariate time series classification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, 2023, pp. 11 497–11 505.

[19] H. Liu, D. Yang, X. Liu, X. Chen, Z. Liang, H. Wang, Y. Cui, and J. Gu, "Todynet: Temporal dynamic graph neural network for multivariate time series classification," *Information Sciences*, vol. 677, p. 120914, Aug. 2024.

[20] K. Yi, Q. Zhang, W. Fan, S. Wang, P. Wang, H. He, N. An, D. Lian, L. Cao, and Z. Niu, "Frequency-domain MLPs are more effective learners in time series forecasting," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[21] J. Duan, W. Zheng, Y. Du, W. Wu, H. Jiang, and H. Qi, "MF-CLR: Multi-frequency contrastive learning representation for time series," in *Forty-first International Conference on Machine Learning*, 2024.

[22] T. Tian, C. Miao, and H. Qian, "Frera: A frequency-refined augmentation for contrastive learning on time series classification," in *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*. ACM, Aug. 2025, p. 2835–2846.

[23] A. Bagnall, H. A. Dau, J. Lines, M. Flynn, J. Large, A. Bostrom, P. Southam, and E. Keogh, "The uea multivariate time series classification archive, 2018," *arXiv preprint arXiv:1811.00075*, 2018.

[24] Y. Mu, M. Shahzad, and X. X. Zhu, "Mptsnet: Integrating multiscale periodic local patterns and global dependencies for multivariate time series classification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, 2025, pp. 19 572–19 580.