arXiv:2312.05698v2 [cs.LG] 9 May 2025

Unsupervised Multi-modal Feature Alignment for Time Series Representation Learning

Chen Liang, Donghua Yang, Zhiyu Liang, Zheng Liang, Xiyang Zhang, Hongzhi Wang, Jianfeng Huang,
Email: 23B903050@stu.hit.edu.cn, yang.dh, zyliang, lz20@hit.edu.cn, 7203610125@stu.hit.edu.cn,
wangzh@hit.edu.cn, JFHuang.research@gmail.com

***Abstract*—Unsupervised representation learning (URL) for time series data has garnered significant interest due to its remarkable adaptability across diverse downstream applications. It is challenging to ensure the utility for downstream tasks by only identifying patterns from the temporal domain features since the goal of URL differs from supervised learning methods. A variety of time series transform techniques, e.g., spectral features, wavelet transformed features, features in image form and symbolic features, etc., transform time series to multi-modal informative features. This study proposes an alignment-based method to harvest transform properties from them. In contrast to common methods that fuse features from multiple modalities, our proposed method simplifies the final neural architecture by retaining a single time series encoder. By validating the proposed method on a diverse set of time series datasets, our approach outperforms existing state-of-the-art URL methods across 3 diverse downstream tasks. This paper introduces a novel model-agnostic paradigm for time series URL, paving a new research direction. Code available at https://anonymous.4open.science/r/MMFA-8E8B.**

***Index Terms*—Time series analysis, unsupervised learning, spectral analysis, classification, anomaly detection, clustering.**

I. Introduction

Time series describes a variable over time encompassing monitoring measures. Multivariate time series (MTS) involves sets of dependent variables, playing a pivotal role across domains from finance to healthcare and the natural sciences [1]–[4].

Unsupervised representation learning for time series. Time series pose complexities in pattern identification and analysis. This further increases the difficulty of obtaining labels for supervised learning [5]. Thus, unsupervised representation learning (URL) for time series [6]–[9] has emerged. *URL aims to train a neural network (called encoder), without requiring labels, to encode the data into geometric signatures (or embedding), to capture inherent patterns of the raw data.* The learned representation, necessitating minimal annotation [10], proves valuable in a variety of downstream tasks including temporal data classification, clustering [11], [12], anomaly detection [8], [9], [13], retrieval [14], [15], etc.

Representation arrangement mechanisms for URL. Moreover, akin to deep clustering [16] and manifold learning [17] methods, URL can help the encoders better capture local [18], [19] or global [20] structures, which can be induced from patterns present in the original samples and learn from prior deductive knowledge. As a result, unsupervised methods with simple classification heads on learned representations may outperform methods tailored for downstream tasks [8].

Transforms as feature engineering for time series. Researchers have explored transforming time series data into image [21], [22] or symbol bags [23], [24] to comprehend them better. Various feature engineering methods are proposed to parse time series and expose abundant informative patterns that are more easily identified. However, the challenges for leveraging these transforms can be further explored as follows.

Challenge 1: Disjoint transform properties of multi-modal features Invariance refers to representation unchanged when input changes, while equivariance refers to representation changing with input, according to certain rules [25]. They are **transform properties** of the transforms and encoders. These transform properties can introduce consistent but disjoint prior knowledge. Learning from them individually introduces gaps between learned transform properties [7]. Besides, the encoder merely needs to identify easily learnable patterns during feature fusing [26], without considering algorithmic associations between different multi-modal patterns, causing learning from shortcuts and blended blind source noises. This can lead to the learning of false patterns [27], resulting in information loss [28], data misinterpretation [29], and decreased model performance [30].

Challenge 2: Coupled transforms and encoders during inference. Feature fusion structures may overly complicate the neural network [7], [22], [24]. Most feature engineering methods for time series are highly coupled with the neural structure of the encoder. For example, a dual tower structure fuses time and spectral domain features, filling the inductive gap between these views. It is necessary to acquire Fourier-transformed features first and encode both features whenever we use the encoder. The cost would be unbearable if more time-consuming transforms and encoders like continuous wavelet transform [31] and a 2D ResNet architecture [21] were involved during inference.

Overcome the challenges. To overcome the above challenges, we inject prior inductive bias implemented by multi-modal transforms into the raw time series encoder and propose a model-agnostic framework. This encoder recovers the most salient local and global arrangement preserved by various transforms and feature extraction processes.

Hongzhi Wang is the corresponding author.

0000–0000/00$00.00 © 2021 IEEE

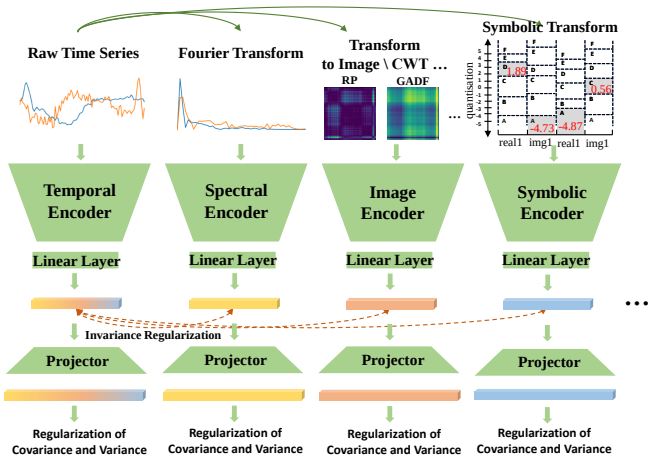


Fig. 1. **An overview of the MMFA framework.** A range of transforms is applied to raw time series to produce informative features across multiple modalities. These features are subsequently processed by neural feature extractors to identify different patterns. Following this, the representations are mapped to an embedding space and aligned via the regularization method.

Methods. We treat the encoders as a graph spectral embedding learner on the graph which aligns the multi-modal features. The alignment with the transforms fine-tunes the global arrangement of the representations, which originally preserves only the local arrangement of the raw time series. Therefore, only preserving the neural encoder for the temporal domain is adequate during inference. Our analysis demonstrates that this framework significantly outperforms state-of-the-art methods on multiple downstream tasks.

The main contributions of this paper are summarized as follows:

- We introduce a novel feature-aligned URL framework, leveraging intricate feature engineering and signal processing methods, outperforming currently available methods while preserving high scalability.
- The proposed method learns local and global arrangement of multi-modal features, and sufficiently leverages different inductive biases introduced.
- The close-formed solutions of the proposed objectives presents recovery of classic manifold learning methods, pointing toward a new model-agnostic and theoretically-oriented URL for TS research direction.
- Through experiments on time series datasets in different domains and with different characteristics (i.e., length, dimensionality, and train set sizes), our proposed approach outperforms all existing URL approaches by a large margin, better than approaches tailored to multiple downstream tasks.

II. Related Work

In this section, we overview the research in the landscape related to the domain of our work. We categorize them into three themes and discuss their contributions, limitations, and relevance to our work.

A. Representation Learning for Time Series

Introducing representation learning for time series Unsupervised representation learning provides a scalable approach to obtain meaningful representations [32]. Neural architectures such as autoencoders and seq2seq models [33] have paved the way for methods based on data reconstruction [34], [35], and context prediction to learn representations [36], [37]. Contrastive learning distinguishes the strict positive (augmented by the same sample) and negative (positive) sample pairs to capture invariance after augmentation, which is presented as an important method [6], [9], [38]–[40]. Reconstruction and generative methods [10], [41], can also produce their hidden states as dense representations. Representation learning not only unifies different tasks but also provides scalable ways for time series data management [15].

Learning arrangements with different global and local levels. Notably, URL encompasses multi-levels of learning objectives: instance level [42]–[44], cluster (prototype) level [45]–[47], and temporal level [5], [6], [48], etc., each capturing different aspects of similarity and dependency in data, representing preservation of *local* or *global* arrangement of the temporal data.

On time series, instance-level contrastive learning perturbs original samples, emphasizing the retention of salient features [9]. In this case, the proposed supervision signal enhances the preservation of the *local* arrangement of data sampled from a distribution of permuted samples. It works well when downstream task goals are sensitive to *local* fluctuations.

Cluster-level contrastive learning extends the focus to the *global* shape of sample distributions [47]. Meanwhile, temporal-level contrastive learning captures temporal dependencies surrounding each sample, learning the causal structure of time series data as *global* arrangement [6]. GRAIL [15] learns temporal representation vectors to reconstruct the distance matrix with the temporal invariance distance measure. These methods *globally* align *local* structures captured naturally by the encoder, thus, injecting different global information into the feature encoder.

B. Feature Transforms

Researchers have investigated various signal processing and feature engineering techniques. These techniques help preserve salient patterns that are challenging to capture without additional prior knowledge. (1). **Discrete Fourier transform (DFT)** [7], [49], which reveals the spectral structure of the original time series. In [7], the authors find that with the same training methods, URL using only DFT features or raw data share a small proportion of false prediction during evaluation, with the rest of them being non-overlapping, revealing relatively complementary inductive bias. (2). **Continuous wavelet transform (CWT)** CWT is a tool that provides an overcomplete representation of a signal by letting the translation and scale parameter of wavelets vary continuously [31]. Compared to raw time series that collapse spectral features, or DFT obscuring temporal stages and trends, Continuous Wavelet Transform (CWT) arranges temporal and spectral patterns in a 2D plane. (3). **Image-like features** (i.e., GAF, RP [21]), which

TABLE I
COMPOSITION OF TRANSFORM OPERATORS AND NEURAL FEATURE EXTRACTORS.

Transform Operator		Feature Extractor	
Raw Time Series		ST	
FFT		ResNet1d	
Image Encoding	PR GADF	ResNet12	
CWT	db1 dmey coif5	ResNet12	
SFA		Longformer	Pre-trained Random

can be encoded leveraging the local precentral view of CNNs. [21] proposes a framework to encode time series data as different types of images thus allowing machines to visually recognize and classify time series. E.g., using a polar coordinate system, Gramian Angular Field (GAF) images a represented as a Gramian matrix where each element is the trigonometric sum between different time intervals. (4). **Symbolic features** Techniques like Symbolic Aggregate Approximation (SAX) [50] or Symbolic Fourier Approximation (SFA) [23] transform noisy time series data into abstract symbolic patterns, filtering out noise and reducing classifier overfitting. SFA, a promising method for time series classification [24], aids in handling high-dimensional, sparse data prone to overfitting.

Generally, these methods provide overcomplete representations that are challenging to fully take advantage of in various tasks. These methods are always can be accelerated with GPU based parallel computation [51], [52].

Our algorithm learns global and local sample arrangements effectively, resulting in stronger theoretical guarantees and outperforming state-of-the-art URL methods.

III. PRELIMINARIES

This section defines the key concepts used in the paper. First, we clarify the data type we study in this paper, multivariate time series. Then we introduce the transforms and encoders with their domain and codomain.

A. Unsupervised Representation Learning for Multivariate Time Series

Multivariate time series is a set of variables, containing observations ordered by successive time. We denote a MTS sample with D variables (a.k.a. dimensions or channels) and T timestamps (a.k.a. length) as $x \in \mathbb{R}^{D\times T}$, and a dataset containing N samples as the $X = \{x_1, x_2, \ldots, x_N\} \in \mathbb{R}^{N\times D\times T}$.

Given an MTS dataset X, the goal of unsupervised representation learning (URL) is to train a neural network model (encoder) $f : \mathbb{R}^{D\times T} \to \mathbb{R}^{d_z}$, where d_z is the dimensionality of the representation, the acquired representation $z_i = f(x_i)$ can be informative for downstream tasks, e.g., classification and forecasting. "Unsupervised" means that the labels of downstream tasks are unavailable when training f. To simplify the notation, we denote $f(X) = \{z_1, z_2, \ldots, z_N\} \in \mathbb{R}^{N\times d_z}$ in following sections.

B. Multi-modal Features and Neural Encoders

Concerning raw MTS data and multi-modal features transformed, we denote the features as $x^{(i)} = T^{(i)}(x)$ in this work, and i denotes the index of the selected transform. The transformed features are highly heterogeneous from the original raw MTS data. Therefore, we designed different neural architectures for the feature encoders w.r.t the multi-modal features, $z^{(i)} = f^{(i)}(x^{(i)})$. They can be composited as $f^{(i)} \circ T^{(i)}$. But the selection of $f^{(i)}$ and $T^{(i)}$ should be compatible. The compositions of them are listed in **Tab.** I.

C. Neural Encoder

In this section, we briefly introduce the encoders for all the modalities transformed from raw time series, and their domains and codomains, which are first mentioned in **Sec.** III-B.

Shapelet Transformer (ST). We use ST [1] as the backbone encoder after unsupervised learning. ST converts time series into energy sequences via learned wavelet-like shapelets, sliding across the signal and computing subsequence distances. By varying shapelet lengths (like wavelet dilation/compression), ST adaptively fits signal shapes and optimizes temporal-spectral resolution, akin to CWT. Here, ST serves as the raw temporal feature extractor $f : \mathbb{R}^{D\times T} \to \mathbb{R}^{d_z}$, processing identity-transformed inputs $T^{(id)} : \mathbb{R}^{D\times T} \to \mathbb{R}^{D\times T}$. Other encoders follow similarly.

CNN & Transformer Feature Extractors. For spectral and 2D patterns, we use ResNet CNNs—1D for DFT-transformed signals ($T^{(dft)} : \mathbb{R}^{D\times T} \to \mathbb{R}^{D\times T}$) and 2D for image-style ($T^{(img)}$) and CWT-based ($T^{(cwt)}$) representations, with interpolation ensuring fixed input dimensions. For symbolic features, we employ a transformer encoder: $T^{(sfa)}$ converts MTS into token sequences (length L) with embeddings fed into a pre-trained transformer, improving downstream performance via sequence modeling. Both architectures adapt to their respective input structures—CNNs for local patterns, transformers for sequential dependencies.

Notations. We denote $X^{(i)} = T^{(i)}(X) = \{x_1^{(i)}, x_2^{(i)}, \ldots, x_N^{(i)}\} \in \mathbb{R}^{N\times \dim T^{(i)}}$ and $Z^{(i)} = f^{(i)}(X^{(i)}) = \{z_1^{(i)}, z_2^{(i)}, \ldots, z_N^{(i)}\}, i \in [0, k]$ in subsequent sections, where $T^{(i)}$, $f^{(i)}$ are ith in k predefined transform and encoder involved and $f^{(0)} = \mathrm{Id}$ is an identity function. Eventually, we discuss the alignment graph and eigenfunctions defined on the sample space, which encompasses not only raw data but also transformed features. The domain of the eigenfunctions can be denoted as $\mathcal{S}^{data} = \sqcup_{i\in[0,k]}\mathrm{Im}(T^{(i)})$, given k transforms, where $\mathrm{Im}(T^{(i)})$ denotes **image/domains of the transforms** which is the space transformed features lies in.

IV. OVERVIEW

In this section, we overview the framework of the proposed multi-modal feature alignment (MMFA).

Spliced images of transforms as entities to be represented. Generally, we treat the transformed features as individual entities in the sample space $\mathcal{S}^{data} = \sqcup_{i\in[0,k]}\mathrm{Im}(T^{(i)})$. This

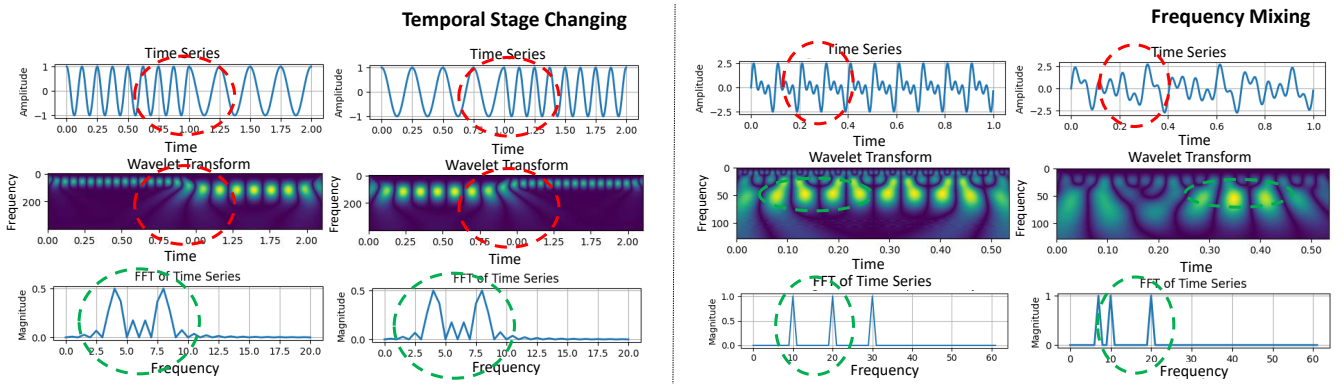


Fig. 2. **Two types of patterns of interest with three multi-modal feature views.** (raw time series, CWT, and FFT of time series) With a green dash circle indicating patterns that are hard to distinguish and, a red dash circle indicating patterns that are easy to identify. The feature encoders take these multi-modal features as input. Difficulties for them to capture certain patterns vary with different transforms, causing different probabilities for them in determining two sample points to be identical.

procedure maps the euclidean space $\mathcal{S}^{data}$ to an generic embedding space $\mathbb{R}^{d_z}$ through a piecewise function $\phi : \mathcal{S}^{data} \to \mathbb{R}^{d_z}$. It can be defined as a piecewise function.

$$\forall i \in [0,k], \forall x \in \mathrm{Im}(T^{(i)}) \qquad \phi(x) = f^{(i)}(x) \tag{1}$$

Arragement of multi-modal entities. For learning ϕ with high utility, leveraging spectral embedding theory, URL relies on an undirected graph $\mathcal{G}(\mathcal{X}, w)$ that indicates sample arrangement lies in the manifold in the original high dimensional space from where raw data is collected, where $\mathcal{X} \in [(k+1)N]$ denotes the subscript for raw MTS and the transformed features. In this paper, the relation matrix $w \in \mathbb{R}^{(k+1)N \times (k+1)N}$ is constructed by connecting entities transformed from the same time series, and similar patterns lie in each of the feature modalities.

$$w_{ij} =: \begin{cases} A_{T_{(\mathcal{X}_i)}, T_{(\mathcal{X}_j)}}(\mathcal{X}_i, \mathcal{X}_j), i \neq j \\ 0, i = j \end{cases} \tag{2}$$

Eq 2 defines a global alignment subgraph $\mathcal{G}$ for low-dimensional reconstruction, where the adjacency function $A_{T_{(\mathcal{X}_i)}, T_{(\mathcal{X}_j)}}$ identifies data points $\mathcal{X}_i, \mathcal{X}_j$ originating from the same sample via transforms $T_{(\mathcal{X}_i)}$ and $T_{(\mathcal{X}_j)}$, which generate modality-specific patterns. Section V-A details two edge types and their roles in shaping representation utility, while Fig. 2 visualizes how transforms globally adjust graph weights through local alignment effects.

Graph embeddings as entities representations. For the extracted feature vectors $\mathcal{Z} \in \mathbb{R}^{(k+1)N \times d_z}$, which is stacked by $Z^{(i)}, i \in [0,k]$. We learn them as graph embeddings of the entities by solving Eq 3, which is encoded from the encoder $\phi_\theta^T(X_{data})$ with parameters θ.

$$\begin{aligned} &\min_{\mathcal{Z}: \mathcal{Z}^T D \mathcal{Z} = I} Tr(\mathcal{Z}^T (D - w) \mathcal{Z}) \\ =&\min_{\theta: \phi_\theta^T(X_{data}) D \phi_\theta(X_{data}) = I} Tr(\phi_\theta^T(X_{data})(D - w)\phi_\theta(X_{data})) \\ &\text{where} \qquad D = \mathrm{diag}(\sum_j w_{ij}) \end{aligned} \tag{3}$$

Generalization to unseen data. This can be solved as a generalized eigenvalue problem [57]. To generalize the discussion to unseen data, we further discuss how to generalize the problem to extrapolate to infinite new samples sampled according to the data distribution.

Preserving time series encoder only. Finally, only the shapelet neural encoder [8] for the raw MTS data is preserved during adaptations for downstream tasks and inferring processes with the following reasons. (1) *Inference efficiency with a simple structure.* (2) *Universality as MTS as primary input.* This raw data feature encoder memorizes algorithmic knowledge from the transforms and encoders. The scalability of the method is also promoted by the small scale of the neural architecture during inference.

V. Regularization Based Multi-Modal Feature Alignment Algorithm

In this section, we aim to build embeddings that can capture all of the inductive biases introduced, which are learned from a global alignment graph $\mathcal{G}$.

Initially, we analyze the approximated eigenfunctions of the graph Laplacian operator, each of which arrages the representations according to the transform properties. Subsequently, we introduce training goals to acquiring representations of the input data from these approximated eigenfunctions. We then design an algorithm to minimize the training objective.

A. Training Objective Approximating GLE

Supervisory goal. The supervisory goal can be seen as learning *graph Laplacian embedding (GLE)* [58], which gives global information about the data manifold. Since we treat the neural encoders as continuous mappings, they automatedly arrange representations according to local fluctuations. As is shown in **Fig.** 2, different transforms provide distinct local properties that would saliently influence the local arrangement. Initially, we establish fundamental concepts closely associated with the GLE problem. Subsequently, we introduce the training objectives.

Extrapolated distribution. Training and testing can be seen as independently and uniformly sample data samples with a

distribution $p'_{data} : \mathcal{S}^{data} \rightarrow \mathbb{R}$ as entities of interest. The training procedure can be seen as sampling jointly from a probability distribution which can approximate and match the equivalence of pairs of the entities, p_{sim}, whose possibility density function (PDF) can be constructed as a piecewise function, $\pi = \frac{1}{k} p_{T_{(x_i)}, T_{(x_j)}}(x_i, x_j) : \mathcal{S}^{data} \times \mathcal{S}^{data} \rightarrow [0, 1]$, where $T_{(x_m)} = T^{(n)}$ denotes the transform used to acquire $x_m \in X^{(n)}$, and $\mathcal{S}^{data} = \sqcup_{i \in [0,k]} \text{Im}(T^{(i)})$ which we fist introduce in **Sec.** III-B. Since the different transform pairs are sampled equally, the joint distributions for them are simply sliced and normalized.

Capturing distributions with eigenfunctions. Next, we can generalize $\mathcal{G}$ into a graph describing any of the entities conforms p'_{data}, and weighted edges between them. Since the GLE problem is strongly related to the concept of the Laplacian operator and its eigenfunction, we next define them in Def. 1 and Def. 2, respectively.

Definition 1. *For a function $f : \mathcal{S}^{data} \rightarrow \mathbb{R}$, the* Laplacian operator $\mathbb{L}$ *for PDF π is defined as:*

$$\begin{aligned} \mathbb{L}(f)(x) &= f(x) + \int_{x'} \frac{\frac{1}{k} p_{T_{(x)}, T_{(x')}}(x, x')}{p'_{data}(x)} f(x') dx' \\ &= f(x) + \frac{1}{k} \sum_{i=1}^{k} \int_{x' \in T^{(i)}(\overline{\mathcal{X}})} \frac{p_{T_{(x)}, T_i}(x, x')}{p'_{data}(x)} f(x') dx' \end{aligned} \tag{4}$$

Definition 2. *An* eigenfunction *of the Laplacian operator $\mathbb{L}$ with eigenvalue λ is a function f satisfies Eq. 5:*

$$\mathbb{E}_{x \sim p'_{data}}[(\mathbb{L}(f)(x) - \lambda f(x))^2] = 0 \tag{5}$$

Generally, eigenfunctions with small eigenvalues would correspond to clusters that are almost disconnected from the rest of the graph, which is well-known in the spectral graph theory [59].

Based on this and the Laplacian operator $\mathbb{L}$ definition, **MMFA** enhances base encoder representations. The chosen transform-encoder compositions construct sub-graphs from raw MTS, each revealing distinct discriminative patterns. The enhancement occurs through two mechanisms. The transform-encoder compositions show the robustness of irrelevant patterns that are introduced by each other. The alignment method we proposed assigns high priority to learning these transform properties.

Reinforcing various subtle invariance and patterns. Some of the important patterns may be ignored by certain compositions but can be enhanced by another one. The added edges help to inject deduced invariance from one composition to the others. These added edges are introduced by π. **Theorem** 1 models more connected components of the graph to eigenvectors by reducing the distance between representations of entities at each end of the edges.

Theorem 1. *Equivalence between Eigenvalues and Distance Reduction of Spectral Embeddings.*

$\mathbb{E}_{(x,x') \sim p_{sim}}[(f(x) - f(x'))^2]$ denotes the expected squared difference between representations of data point pairs under distribution p_{sim}, and f is an eigenfunction of $\mathbb{L}$. Eq. 6 demonstrates the relationship as follows:

$$\mathbb{E}_{(x,x') \sim p_{sim}}[(f(x) - f(x'))^2] = 2\lambda \mathbb{E}_{x \sim p'_{data}}[f(x)^2] \tag{6}$$

Proof of the theorem is in **Sec.** VIII-A.

Remark 1. ***Theorem*** *1 illustrates that a lower eigenvalue λ correlates with reduced distance between nearly aligned pairs. Consequently, minimizing the distance during training aligns the encoder with the eigenfunctions of the Laplacian operator possessing small eigenvalues, effectively segregating the graph-captured relations into relatively distinct clusters.*

We denote raw MTS samples and their distribution as $x \sim p_{data}$. X is a set of samples sampled with p_{data}. To capture the global invariance implied in w, the corresponding loss can be estimated and bounded by Inequation 7, where $Z \in \mathbb{R}^{(k+1) \times N \times d}$ which is reshaped from $\mathcal{Z}$, and $z_i^{(m)} \in Z^{(m)}$ denotes the representation for the ith sample transformed and encoded by $T^{(m)}$ and $f^{(m)}$.

To ensure the feasibility of training, we provide an upper bound of the originally proposed loss function to reduce the training workload, which is demonstrated at **Theorem** 2.

Theorem 2. *Multi-modal Invariance Estimation.*

$x \sim p_{data}$ denotes raw Multivariate Time Series (MTS) samples, where X represents a group of samples sampled with p_{data}. Strictly, $p_{T^{(i)}, T^{(j)}}(T^{(i)}(x), T^{(j)}(x)) = 1, x \in X$. This captures the global equivariance implied in w, the corresponding loss, denoted by $\mathcal{L}'_{inv}(Z)$, can be estimated and bounded as shown in Inequality 7.

$$\begin{aligned} \mathcal{L}'_{inv}(Z) &= \frac{1}{Nk(k+1)} \sum_{\substack{m=0 \\ n=0 \\ m \neq n}}^{k} \sum_{i=1}^{N} ||z_i^{(m)} - z_i^{(n)}||_2^2 \\ &\leq \frac{2}{N(k+1)} \sum_{i=1}^{N} \sum_{m=1}^{k} ||z_i^{(0)} - z_i^{(m)}||_2^2 \end{aligned} \tag{7}$$

*Proof of the theorem is in **Sec.** VIII-B.*

The expression $\mathcal{L}'_{\text{inv}}(Z)$ simplifies the optimization objective aimed at capturing global equivariance among the multi-modal features.

According to the **Theorem** 2, we define $\mathcal{L}_{inv}(Z)$ as the optimization objective capturing global invariance among the multi-modal features.

$$\mathcal{L}_{inv}(Z) = \frac{1}{N(k+1)} \sum_{i=1}^{N} \sum_{m=1}^{k} ||z_i^{(0)} - z_i^{(m)}||_2^2 \tag{8}$$

Theorem 3. *Orthogonality of Eigenfunction-Based Representations.*

For $\forall g, h : \mathbb{R}^{data} \rightarrow \mathbb{R}$, $\mathbb{E}x \sim p'_{data}[\mathbb{L}(g)(x) \cdot h(x)] = \mathbb{E}x \sim p'_{data}[g(x) \cdot \mathbb{L}(h)(x)]$. Therefore, $\mathbb{L}$ is a symmetric linear operator. The representations are outputs derived from approximated eigenfunctions of $\mathbb{L}$ possessing low eigenvalues, $f_{rep}(\cdot) = [f_1(\cdot), f_2(\cdot), \ldots, f_{d_z}(\cdot)]^T$, and $\mathbb{L}$ is a symmetric real value linear operator. According to spectral theorem [59], it is feasible to ensure orthogonality while maximizing the overall information:

Algorithm 1 Asymmetric Neural Encoders Alignment

Input: D: MTS dataset; $T^{(i)}, i \in [k]$: transform operators; $f^{(i)}, i \in [k]$: neural encoder architectures.
Output: f: Trained ST encoder for MTS representation.
$\theta_f \leftarrow$ Initialize parameters of the raw MTS encoder.
for $i \in [k]$ **do**
  $\theta_{f^{(i)}} \leftarrow$ Initialize parameters of the encoders w.r.t $T^{(i)}$.
end for
while not converged **do**
  $X \leftarrow$ Sample a mini-batch from D.
  for $i \in [k]$ **do**
    $X^{(i)} \leftarrow T^{(i)}(X)$ # Transform raw MTS.
    $\theta_{\tilde{f}}, \theta_{\tilde{f}^{(i)}} \leftarrow U(f, f^{(i)}, X, X^{(i)})$ # Accumulate gradients.
    $\theta_{f^{(i)}} \leftarrow \theta_{f^{(i)}} + \epsilon \frac{1}{k}(\theta_{\tilde{f}^{(i)}} - \theta_{f^{(i)}})$ # Update encoder.
  end for
  $\theta_f \leftarrow \theta_f + \epsilon(\theta_{\tilde{f}} - \theta_f)$ # Update encoder.
end while
Return: f

$$\mathbb{E}_{x \sim p'_{data}}[f_{rep}(x) f_{rep}(x)^T] = \mathbb{I} \tag{9}$$

According to **Theorem** 3, we design two relevant losses, $\mathcal{L}_{var}$ and $\mathcal{L}_{cov}$, in Eq. 10 and Eq. 11, respectively.

$$\begin{aligned} &\mathcal{L}_{cov}(Z) \\ &= \frac{1}{(N-1)kd} \sum_{l=1}^{k} \sum_{m \neq n} [\sum_{i=1}^{N} (z_i^{(l)} - \bar{z}^{(l)})(z_i^{(l)} - \bar{z}^{(l)})^T]_{m,n} \\ &where\ \bar{z}^{(l)} = \frac{1}{n} \sum_{i=1}^{n} z_i^{(l)} \end{aligned} \tag{10}$$

$$\mathcal{L}_{var}(Z) = \frac{1}{dk} \sum_{l=0}^{k} \sum_{j=1}^{d} max(0, 1 - \sqrt{Var(\{z_{i,j}^{(l)} | i \in [N]\}) + \epsilon}) \tag{11}$$

According to Eq. 8, Eq. 10 and Eq. 11, the end-to-end training object is obtained in equation 12. α, β and γ are empirical parameters.

$$\mathcal{L} = \alpha \mathcal{L}_{cov} + \beta \mathcal{L}_{var} + \gamma \mathcal{L}_{inv} \tag{12}$$

Spectral selection mechanism ensures the balance of transforms as information sources. Our further discussion deduced the closed-form solutions of linear degenerated **MMFA** with $k = 0$ in **Lemma** 1, $k = 1$ in **Lemma** 2, and $k = p > 2$ in **Lemma** 3 transforms aligned with the identity transformation in **Sec.** VIII-D. According to the closed-form optimal solutions, we can observe that **MMFA** serves as a spectral selection mechanism for balancing variances of empirical distributions measured with the transformed features.

Theorem 4. *The optimization problem of **MMFA** recovers kernel principal component analysis (KPCA) [60] and kernel canonical correlation analysis (KCCA) [61].*

Proof of the theorem is in **Sec.** VIII-E.

B. Asymmetric Encoders Alignment Optimizing Algorithm

We finally design Algorithm 1 minimizing the training objective by mini-batch gradient descent. Algorithm 1 takes the MTS dataset D, a set of transform operators $T^{(i)}$, and corresponding neural architecture for the encoders as input. Then the parameters of the encoders f and $f^{(i)}$ are initialized in lines 1-3. In lines 7-9, update step U is defined as $U(f, f^{(i)}, X, X^{(i)}) = \theta - \nabla_\theta \mathcal{L}([f^{(i)}(X^{(i)}), f(X)])$, where θ denotes the parameters of the two encoders. We divide the designed training objective into k parts, iteratively computes and accumulates gradient for both f and $f^{(i)}$. After accumulating all the gradients of the batch, the encoder f is updated.

After the training phase, we only need to preserve the encoder, f, for raw MTS and adapt it to diverse downstream tasks.

VI. Empirical Analysis

In this section we provide empirical evidence to justify our main claims. To ensure throughness and fairness of the analysis, we provide details about the implementations of each benchmarks and baselines in **Sec.** VII

Extensive experiments on various datasets. In this section, we conduct extensive experiments containing 31 real-world datasets for three important downstream tasks, classification, clustering, and anomaly detection. The 30 datasets for classification and 12 of them for clustering vary with their **training set sizes**, **dimensionalities**, **lengths**, and **numbers of classes** which are detailed in **Tab.** VIII. The remaining 1 datasets for anomaly detection have different **numbers of entities**, **dimensionality**, **length**, and **anomaly ratio** which are detailed in **Tab.** IX. Standard metrics are used for various tasks, i.e., Accuracy (Acc) [65] for classification, Cohen's d [78] for standardizing performance on various datasets, RI [80] and NMI [81] for clustering, and F1-score [62] for anomaly detection.

Baselines with significant diversity. The baselines are divided into two groups,i.e., the URL methods and methods tailored to specific tasks. All the URL methods share the same evaluation protocol on downstream tasks, by feeding the representations to classical models, i.e., SVM [82], IF [83], and KMeans [84]. The tailored methods are evaluated following the original implementations. More detailed demonstrations of the downstream tasks, datasets, evaluation metrics, and baselines are elaborated in **Sec.** VII.

Comprehensive research questions. We evaluate the robustness and adaptability of **MMFA** across diverse datasets and domains. Through experiments, we aim to scrutinize algorithms, assess scalability, and compare them to the proposed **MMFA** framework, outlined by our research questions (RQs).

RQ 1. *How does the **MMFA** framework perform compared to other unsupervised methods?*

Tab. II, III, and IV compare classification, clustering, and anomaly detection results. The proposed **MMFA** consistently outperforms unsupervised baselines across tasks, achieving top

TABLE II
COMPARISONS OF PERFORMANCE IN MTS CLASSIFICATION, EVALUATED ON ACCURACY SCORES. THE TOP-PERFORMING RESULTS OF URL METHODS ARE SHOWN IN BOLD, WITH † DENOTING THE OVERALL BEST PERFORMANCE.

Dataset	**Tailored Classification Approaches**						**Unsupervised Representation Learning + Classifier**							
	DTWD	MLSTM-FCNs	TapNet	ShapeNet	OSCNN	DSN	TS2Vec	T-Loss	TNC	TS-TCC	TST	CSL	MMFA-aug	**MMFA**
ArticularyWordRecognition	0.987	0.973	0.987	0.987	0.988	0.984	0.987	0.943	0.973	0.953	0.977	0.990	0.973	**0.993**†
AtrialFibrillation	0.200	0.267	0.333	0.400	0.233	0.067	0.200	0.133	0.133	0.267	0.067	0.533	0.533	**0.600**†
BasicMotions	0.975	0.950	1.000†	1.000†	1.000†	1.000†	0.975	**1.000**†	0.975	**1.000**†	0.975	**1.000**†	**1.000**†	**1.000**†
CharacterTrajectories	0.989	0.985	0.997	0.980	0.998†	0.994	**0.995**	0.993	0.967	0.985	0.975	0.991	0.987	0.991
Cricket	1.000†	0.917	0.958	0.986	0.993	0.989	0.972	0.972	0.958	0.917	**1.000**†	0.994	0.986	**1.000**†
DuckDuckGeese	0.600	0.675	0.575	0.725†	0.540	0.568	**0.680**	0.650	0.460	0.380	0.620	0.380	0.620	**0.680**
EigenWorms	0.618	0.504	0.489	0.878†	0.414	0.391	**0.847**	0.840	0.840	0.779	0.748	0.779	0.768	0.840
Epilepsy	0.964	0.761	0.971	0.987	0.980	0.999†	0.964	0.971	0.957	0.957	0.949	0.986	0.899	**0.987**†
ERing	0.133	0.133	0.133	0.133	0.882	0.922	0.874	0.133	0.852	0.904	0.874	**0.967**†	0.922	**0.967**†
EthanolConcentration	0.323	0.373	0.323	0.312	0.241	0.245	0.308	0.205	0.297	0.285	0.262	0.498	0.331	**0.551**†
FaceDetection	0.529	0.545	0.556	0.602	0.575	0.635†	0.501	0.513	0.536	0.544	0.534	**0.593**	0.526	0.555
FingerMovements	0.530	0.580	0.530	0.580	0.568	0.492	0.480	0.580	0.470	0.460	0.560	0.590	0.490	**0.610**†
HandMovementDirection	0.231	0.365	0.378	0.338	0.443	0.373	0.338	0.351	0.324	0.243	0.243	0.432	0.351	**0.487**†
Handwriting	0.286	0.286	0.357	0.451	0.668†	0.337	0.515	0.451	0.249	0.498	0.225	**0.533**	0.370	0.472
Heartbeat	0.717	0.663	0.751	0.756	0.489	0.783†	0.683	0.741	0.746	0.751	0.746	0.722	0.720	**0.761**
InsectWingbeat	N/A	0.167	0.208	0.250	0.667†	0.386	0.466	0.156	**0.469**	0.264	0.105	0.256	0.425	**0.469**
JapaneseVowels	0.949	0.976	0.965	0.984	0.991†	0.987	0.984	**0.989**	0.978	0.930	0.978	0.919	0.960	0.978
Libras	0.870	0.856	0.850	0.856	0.950	0.964†	0.867	0.883	0.817	0.822	0.656	**0.906**	0.833	0.894
LSST	0.551	0.373	0.568	0.590	0.413	0.603	0.537	0.509	0.595	0.474	0.408	0.617	0.372	**0.622**†
MotorImagery	0.500	0.510	0.590	0.610†	0.535	0.574	0.510	0.580	0.500	0.610	0.500	0.610	0.510	**0.630**†
NATOPS	0.883	0.889	0.939	0.883	0.968	0.978†	0.928	0.917	0.911	0.822	0.850	0.878	0.900	**0.933**
PEMS-SF	0.711	0.699	0.751	0.751	0.760	0.801	0.682	0.676	0.699	0.734	0.740	0.827	0.827	**0.838**†
PenDigits	0.977	0.978	0.980	0.977	0.986	0.987	0.989	0.981	0.979	0.974	0.560	**0.990**†	0.979	0.980
PhonemeSpectra	0.151	0.110	0.175	0.298	0.299†	0.320	0.233	0.222	0.207	0.252	0.085	**0.255**	0.183	0.216
RacketSports	0.803	0.803	0.868	0.882†	0.877	0.862	0.855	0.855	0.776	0.816	0.809	**0.882**†	0.815	0.862
SelfRegulationSCP1	0.775	0.874	0.652	0.782	0.835	0.717	0.812	0.843	0.799	0.823	0.754	0.846	0.887	**0.901**†
SelfRegulationSCP2	0.539	0.472	0.550	0.578	0.532	0.464	0.578	0.539	0.550	0.533	0.550	0.496	0.617	**0.628**†
SpokenArabicDigits	0.963	0.990	0.983	0.975	0.997†	0.991	0.988	0.905	0.934	0.970	0.923	**0.990**	0.980	**0.990**
StandWalkJump	0.200	0.067	0.400	0.533	0.383	0.387	0.467	0.333	0.400	0.333	0.267	**0.667**†	**0.667**†	**0.667**†
UWaveGestureLibrary	0.903	0.891	0.894	0.906	0.927	0.916	0.906	0.875	0.759	0.753	0.575	0.922	0.922	**0.931**†
Avg Cohen's d Effect														
incl. InsectWingbeat	/	-0.670	-0.024	0.366	0.357	0.280	0.089	-0.176	-0.450	-0.447	-0.860	0.502	0.096	**0.937**†
excl. InsectWingbeat	-0.376	-0.631	0.033	0.434	0.321	0.308	0.094	-0.116	-0.473	-0.425	-0.825	0.567	0.109	**0.979**†
Avg Ranking (URL only)	/	/	/	/	/	/	3.96	4.63	5.40	5.30	6.20	2.83	4.43	**1.60**
Avg Ranking (All)	9.40	9.43	6.90	5.43	5.37	5.83	6.83	7.90	9.37	9.10	10.46	4.40	7.17	**2.63**†

accuracy in 21/30 datasets. A variant using only basic data augmentations (MMFA-aug) underperforms significantly, as **MMFA**'s integration of diverse transforms and neural encoders extracts denoised, discriminative patterns (Sec. V).

In clustering, **MMFA** excels except on PenDigits (too short sequences) and AtrialFibrillation. For anomaly detection, it outperforms competitors by large margins, producing compact representation boundaries through multi-modal feature alignment.

By leveraging algorithmic biases from multi-modal transforms rather than simple augmentations, **MMFA** induces robust global structures, surpassing methods reliant on limited feature engineering.

RQ 2. *How does the **MMFA** framework perform compared to the supervised approaches and approaches tailored for the downstream tasks?*

The comparison between **MMFA** and tailored discriminative models on classification tasks is shown in **Tab.** II. **MMFA** outperforms tailored fully supervised competitors for time series classification on 16 UCR classification datasets, indicating its extraction of richer information directly from raw time series during inference.

According to **Tab.** III, over all of the clustering tasks, among the tailored methods, only MUSLA [76] is compatible with **MMFA**. **MMFA** also shows significant advantages among the URL methods.

In contrast to the compared tailored methods in **Tab.** II and **Tab.** III, unsupervised methods learn to transform properties of data without task-specific tuning, enhancing feature extraction capabilities that go beyond specific tasks, and fostering potential generalizability.

RQ 3. *What is the impact of data characteristics?*

In our analysis, the evaluated datasets exhibit significant variations in training dataset sizes, numbers of channels, time series length, and salient patterns. **Tab.** VIII illustrates the diversity: training set sizes range from 12 (StandWalkJump, comprising 3 classes with only 4 samples per class, constituting a 3-way, 4-shot learning task) to 30,000 (InsectWingbeat, characterized by abundant training samples). Consequently, we conduct correlation analysis on the UEA datasets focusing on these three data characteristics and show the findings in **Fig.** 3.

As depicted in (a) of **Fig.** 3, **MMFA** exhibits a high effect size on performance improvement with smaller training sets. This observation suggests that **MMFA** excels in achieving higher few-shot performance.

A slight negative correlation exists between classification performance improvement effect size and dataset dimensionality, which shows **MMFA** outperforms more significantly on low dimensional datasets, showcasing **MMFA**'s strong generalization effects that avoid overfitting with certain patterns with limited input features.

Conversely, positive correlations between effect size and time series length indicate **MMFA**'s stronger adaptability to longer time series than baselines. However, a limitation is

TABLE III
THE MTS CLUSTERING PERFORMANCE EVALUATED ON RI [80] AND NMI [81], WITH THE MOST EFFECTIVE RESULTS FROM URL METHODS IN BOLD, WITH † INDICATING SUPERIORITY OVER OTHERS.

Dataset	Metric	Tailored Clustering Approaches							Unsupervised Representation Learning + Clustering							
		MC2PCA	TCK	m-kAVG+ED	m-kDBA	DeTSEC	MUSLA	Time2Feat	TS2Vec	T-Loss	TNC	TS-TCC	TST	CSL	MMFA-aug	**MMFA**
ArticularyWordRecognition	RI	0.989	0.973	0.952	0.934	0.972	0.977	0.977	0.980	0.975	0.938	0.946	0.978	0.990	0.990	**0.998**†
	NMI	0.934	0.873	0.834	0.741	0.792	0.838	0.881	0.880	0.842	0.565	0.621	0.866	0.942	0.942	**0.983**†
AtrialFibrillation	RI	0.514	0.552	0.705	0.686	0.629	0.724	0.600	0.465	0.469	0.518	0.469	0.444	**0.743**†	0.524	0.648
	NMI	0.514	0.191	0.516	0.317	0.293	0.538	0.261	0.080	0.149	0.147	0.164	0.171	**0.587**†	0.251	0.305
BasicMotions	RI	0.791	0.868	0.772	0.749	0.717	1.000†	0.883	0.854	0.936	0.719	0.856	0.844	**1.000**	0.845	**1.000**
	NMI	0.674	0.776	0.543	0.639	0.800	1.000†	0.833	0.820	0.871	0.394	0.823	0.810	**1.000**	0.806	**1.000**
Epilepsy	RI	0.613	0.786	0.768	0.777	0.840	0.816	0.897†	0.706	0.705	0.650	0.736	0.718	0.873	0.679	**0.876**
	NMI	0.173	0.534	0.409	0.471	0.346	0.601	0.814†	0.312	0.306	0.156	0.451	0.357	0.705	0.530	**0.721**
ERing	RI	0.756	0.772	0.805	0.775	0.770	0.887	0.841	0.925	0.885	0.764	0.821	0.867	0.968	0.968	**0.972**†
	NMI	0.336	0.399	0.400	0.406	0.392	0.722	0.641	0.775	0.672	0.346	0.478	0.594	0.906	0.906	**0.918**†
HandMovementDirection	RI	0.627	0.635	0.697	0.685	0.628	0.719†	0.577	0.609	0.599	0.613	0.608	0.607	0.651	0.630	**0.657**
	NMI	0.067	0.103	0.168	0.265	0.112	0.398†	0.062	0.044	0.034	0.051	0.053	0.039	0.175	0.118	**0.185**
Libras	RI	0.892	0.917	0.911	0.913	0.907	0.941	0.926	0.904	0.922	0.896	0.881	0.886	0.941	0.938	**0.944**†
	NMI	0.577	0.620	0.622	0.622	0.602	0.724	0.703	0.542	0.654	0.464	0.373	0.400	0.761	0.730	**0.785**†
NATOPS	RI	0.882	0.833	0.853	0.876	0.714	0.976†	0.777	0.817	0.836	0.700	0.792	0.809	0.876	0.850	**0.919**
	NMI	0.698	0.679	0.643	0.643	0.043	0.855†	0.482	0.523	0.558	0.222	0.513	0.565	0.657	0.628	**0.864**
PEMS-SF	RI	0.424	0.191	0.817	0.755	0.806	0.892†	0.818	0.765	0.746	0.763	0.789	0.726	0.858	0.842	**0.876**
	NMI	0.011	0.066	0.491	0.402	0.425	0.614†	0.495	0.290	0.102	0.278	0.331	0.026	0.537	0.528	**0.583**
PenDigits	RI	0.929	0.922	0.935	0.881	0.885	0.946	0.914	0.941	0.936	0.873	0.857	0.818	**0.950**	0.921	0.932
	NMI	0.713	0.693	0.738	0.605	0.563	0.826†	0.705	0.776	0.749	0.537	0.339	0.090	**0.822**	0.703	0.730
StandWalkJump	RI	0.591	0.762	0.733	0.695	0.733	0.771†	0.500	0.410	0.410	0.457	0.589	0.467	0.724	0.591	**0.761**
	NMI	0.350	0.536	0.559	0.466	0.556	0.609†	0.077	0.213	0.213	0.193	0.187	0.248	0.554	0.336	**0.611**
UWaveGestureLibrary	RI	0.883	0.913	0.920	0.893	0.879	0.913	0.863	0.865	0.893	0.817	0.796	0.779	0.927	0.927	**0.929**†
	NMI	0.570	0.710	0.713	0.582	0.558	0.728	0.610	0.511	0.614	0.322	0.215	0.244	0.731	0.731	**0.749**†
Avg Cohen's d Effect	RI	-0.451	-0.129	0.287	-0.065	-0.230	1.096	-0.012	-0.227	-0.111	-1.095	-0.771	-0.816	1.084	0.321	**1.119**†
	NMI	-0.244	0.000	0.217	0.004	-0.316	1.152†	0.153	-0.265	-0.232	-1.302	-0.910	-0.892	1.071	0.478	**1.089**
Avg Ranking (URL only)		/	/	/	/	/	/	/	5.08	5.25	6.92	6.08	5.92	1.75	3.25	**1.33**†
Avg Ranking (All)		9.92	7.83	6.75	8.50	9.42	2.92	8.17	9.25	8.92	12.67	11.25	11.67	2.66	6.33	**2.42**†

TABLE IV
COMPARISON OF PERFORMANCE OF ON UCR ANOMALY DETECTION DATASET. THE SLIDING WINDOW'S LENGTH IS SET TO 256, WITH THE MOST OPTIMAL OUTCOMES HIGHLIGHTED IN BOLD.

Methods on UCR	IF-s	TST	TS-TCC	T-Loss	TS2Vec	CSL	**MMFA**
Precision	0.9778	0.7598	0.7863	0.7986	0.8147	0.8040	0.9218
Recall	0.0069	0.3481	0.3431	0.3556	0.3551	0.3664	0.4717
F1	0.0137	0.4775	0.4777	0.4921	0.4946	0.5034	**0.6241**
Accuracy	0.0069	0.9881	0.9877	0.9882	0.9881	0.9886	**0.9920**

that **MMFA** struggles with shorter time series (e.g., PenDigits datasets with length $= 8$), evident in lower performance across classification and clustering tasks. This limitation may stem from the transforms nature, requiring optimal performance from a time series of specific lengths.

RQ 4. *Ablation (Ab) and leave one out (LOO) study.*

According to Table V, more transformations lead to better performance and lower variation in the magnitude of the overall effect. This is the evidence for the robustness of the model.

Transforms show non-additive effects. RP and the DFT show higher performance. RP contributes more independently to performance improvements. The insignificance of DFT in the LOO study suggests that it may be represented by or diminishing other jointly aligned transforms. CWT with CGau2 shows less significance but higher effects in the LOO study.

RQ 5. *Sensitivity analysis for α, β and γ in the trainin objective.*

According to **Tab.** VI and **Tab.** VI, the accuracy demonstrates moderate sensitivity to parameter variations, with optimal $alpha$=10.0 for SelfRegulationSCP2 (0.544) and $alpha$=0.1 for Heartbeat (0.732). For $\beta = \gamma$ parameters, both datasets achieve peak performance at $\beta = \gamma$=0.1/10.0

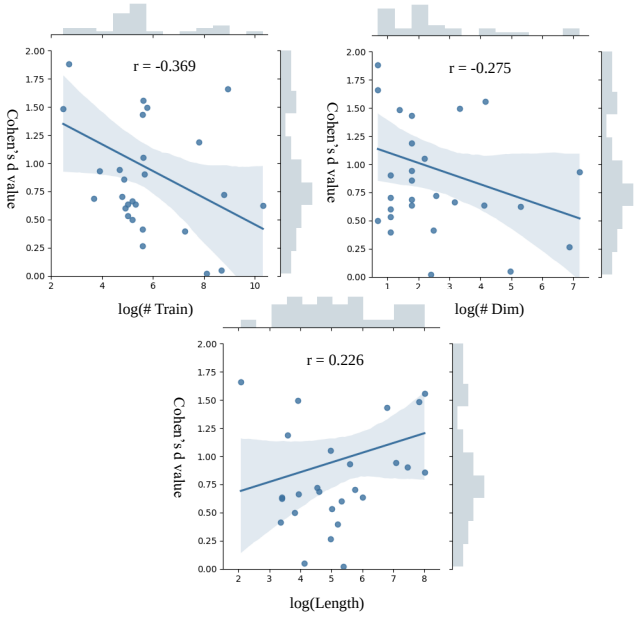


Fig. 3. **Correlation plot** illustrating the relationships between Cohen's d effect sizes of performance improvement made by **MMFA** on 30 UEA datasets and characteristics of the datasets, i.e., training size, time series dimensionality, and time length. Logarithmic transforms are employed to enhance the linearity of the data.

(Heartbeat: 0.732) and $\beta = \gamma$=10.0/25.0 (SelfRegulationSCP2: 0.544), suggesting parameter robustness in mid-to-high ranges, which encourages larger punishment of forbidding orthogonality of the representations.

VII. EXPERIMENT DETAILS

In this section, we exhibit the details of implementations of downstream tasks and baselines, characteristics of the datasets, evaluation metrics, and implementation details of our proposed

TABLE V
ABLATION (AB) AND LEAVE ONE OUT (LOO) STUDY.

Dataset	CGau2	RP	GADF	DFT	WEASEL	CGau2-loo	RP-loo	GADF-loo	DFT-loo	WEASEL-loo
ArticularyWordRecognition	0.983	0.983	0.980	0.980	0.980	0.983	0.980	0.983	0.987	0.983
AtrialFibrillation	0.533	0.533	0.533	0.467	0.467	0.533	0.533	0.533	0.533	0.533
BasicMotions	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
CharacterTrajectories	0.991	0.987	0.987	0.991	0.989	0.990	0.991	0.988	0.990	0.991
Cricket	0.958	0.944	0.958	0.944	0.958	0.972	0.972	0.944	0.958	0.972
DuckDuckGeese	0.300	0.360	0.340	0.340	0.360	0.320	0.300	0.280	0.340	0.460
EigenWorms	0.580	0.550	0.557	0.595	0.557	0.626	0.626	0.588	0.580	0.580
Epilepsy	0.964	0.942	0.978	0.957	0.964	0.957	0.971	0.971	0.964	0.971
EthanolConcentration	0.449	0.479	0.479	0.475	0.426	0.616	0.620	0.631	0.601	0.643
ERing	0.956	0.959	0.952	0.956	0.956	0.952	0.948	0.948	0.944	0.956
FaceDetection	0.564	0.570	0.562	0.564	0.576	0.575	0.573	0.567	0.574	0.568
FingerMovements	0.510	0.490	0.520	0.540	0.530	0.580	0.590	0.590	0.570	0.550
HandMovementDirection	0.405	0.432	0.432	0.432	0.446	0.446	0.392	0.419	0.432	0.486
Handwriting	0.487	0.500	0.496	0.492	0.489	0.474	0.485	0.476	0.481	0.487
Heartbeat	0.722	0.741	0.722	0.732	0.737	0.707	0.722	0.741	0.732	0.722
InsectWingbeat	0.262	0.425	0.466	0.466	0.290	0.262	0.425	0.466	0.466	0.290
JapaneseVowels	0.892	0.895	0.886	0.886	0.895	0.905	0.897	0.900	0.908	0.889
Libras	0.872	0.872	0.872	0.872	0.872	0.889	0.878	0.883	0.894	0.883
LSST	0.607	0.600	0.601	0.603	0.599	0.593	0.603	0.595	0.607	0.601
MotorImagery	0.510	0.520	0.640	0.630	0.610	0.620	0.590	0.600	0.590	0.590
NATOPS	0.817	0.839	0.833	0.822	0.817	0.856	0.811	0.844	0.828	0.828
PenDigits	0.984	0.985	0.985	0.985	0.986	0.984	0.985	0.986	0.986	0.985
PEMS-SF	0.809	0.798	0.798	0.803	0.798	0.803	0.809	0.809	0.809	0.809
PhonemeSpectra	0.241	0.236	0.245	0.236	0.245	0.233	0.222	0.228	0.232	0.225
RacketSports	0.855	0.862	0.855	0.882	0.855	0.895	0.888	0.901	0.901	0.888
SelfRegulationSCP1	0.782	0.785	0.785	0.785	0.782	0.819	0.816	0.812	0.816	0.795
SelfRegulationSCP2	0.467	0.539	0.472	0.472	0.456	0.506	0.506	0.517	0.500	0.483
SpokenArabicDigits	0.980	0.983	0.981	0.977	0.978	0.984	0.983	0.980	0.985	0.985
StandWalkJump	0.533	0.533	0.533	0.533	0.467	0.400	0.533	0.533	0.533	0.533
UWaveGestureLibrary	0.912	0.916	0.912	0.916	0.916	0.912	0.912	0.916	0.906	0.916
Average Effect Size	-0.368	-0.132	-0.239	-0.208	-0.256	0.091	0.113	0.203	0.443	0.352

TABLE VI
ACCURACY COMPARISON WITH DIFFERENT α VALUES.

Dataset	$\alpha = 0.0$	$\alpha = 0.1$	$\alpha = 1.0$	$\alpha = 10.0$	$\alpha = 25.0$
SelfRegulationSCP2	0.528	0.528	0.528	0.544	0.533
Heartbeat	0.712	0.732	0.712	0.722	0.712

TABLE VII
ACCURACY COMPARISON WITH DIFFERENT $\beta = \gamma$ VALUES.

Dataset	$\beta = \gamma = 0.0$	$\beta = \gamma = 0.1$	$\beta = \gamma = 1.0$	$\beta = \gamma = 10.0$	$\beta = \gamma = 25.0$
SelfRegulationSCP2	0.539	0.539	0.528	0.544	0.544
Heartbeat	0.717	0.732	0.722	0.732	0.722

methods, which are sufficient support for our demonstration of the experiments in **Sec.** VI.

A. Experimental Settings

1) Baselines: We use 21 baselines for comparison, which are divided into two groups:

URL methods. We compare our **MMFA** framework with 6-time series URL baselines, including TS2Vec [9], T-Loss [39], TNC [5], TS-TCC [6], TST [10] and CSL [8]. All URL competitors are evaluated similarly to **MMFA** for a fair comparison.

Methods tailored to specific tasks. We include DTWD [65] (1-NN classifier with dynamic time warping) and supervised methods MLSTM-FCNs [66] (RNNs), TapNet [67] (attentional prototypes), ShapeNet [68] (shapelets), OSCNN [69], and DSN [70] (CNNs). Ensemble methods [71] are excluded for fairness. Supervised methods leverage true labels for feature learning, analogous to URL data augmentation, ensuring fair comparison with **MMFA**.

TABLE VIII
STATISTICS ON THE 30 UEA DATASETS ARE EMPLOYED FOR CLASSIFICATION ASSESSMENT, WHILE 12 SPECIFIC SUBSETS (DENOTED BY *) UNDERGO CLUSTERING EVALUATION AS PER THE METHODOLOGY OUTLINED IN [76].

Dataset	# Train	# Test	# Dim	Length	# Class
ArticularyWordRecognition*	275	300	9	144	25
AtrialFibrillation*	15	15	2	640	3
BasicMotions*	40	40	6	100	4
CharacterTrajectories	1422	1436	3	182	20
Cricket	108	72	6	1197	12
DuckDuckGeese	50	50	1345	270	5
EigenWorms	128	131	6	17984	5
Epilepsy*	137	138	3	206	4
EthanolConcentration	261	263	3	1751	4
ERing*	30	270	4	65	6
FaceDetection	5890	3524	144	62	2
FingerMovements	316	100	28	50	2
HandMovementDirection*	160	74	10	400	*4*
Handwriting	150	850	3	152	26
Heartbeat	204	205	61	405	2
InsectWingbeat	30000	20000	200	30	10
JapaneseVowels	270	370	12	29	9
Libras*	180	180	2	45	15
LSST	2459	2466	6	36	14
MotorImagery	278	100	64	3000	2
NATOPS*	180	180	24	51	6
PenDigits*	7494	3498	2	8	10
PEMS-SF*	267	173	963	144	7
Phoneme	3315	3353	11	217	39
RacketSports	151	152	6	30	4
SelfRegulationSCP1	268	293	6	896	2
SelfRegulationSCP2	200	180	7	1152	2
SpokenArabicDigits	6599	2199	13	93	10
StandWalkJump*	12	15	4	2500	3
UWaveGestureLibrary*	120	320	3	315	8

Task-specific benchmarks include the DTWD baseline (1-nearest-neighbor classifier with DTW distance) and supervised methods spanning RNN-based, attention-based, shapelet-driven, and CNN architectures. Ensemble methods were ex-

TABLE IX
STATISTICS OF EVALUATED ANOMALY DETECTION DATASETS.

Dataset	# Entity	# Dim	Train length	Test length	Anomaly ratio (%)
UCR	250	1	5302449	12919799	0.38

cluded for fairness. While supervised approaches leverage true labels—analogous to URL's data augmentation—the comparison remains valid as both paradigms align label-informed feature learning.

We assess six sophisticated clustering benchmarks, including feature selection based Time2Feat [12], dimension-reduction-based MC2PCA [72] and TCK [73], distance-based m-kAVG+ED and m-kDBA [74], deep learning-based DeTSEC [75], and shapelet-based MUSLA [76].

Since no documented anomaly detection evaluations in segment-level settings exist, we create two raw MTS baselines using Isolation Forest for fair comparisons, models operating at each timestamp (IF-p) or within each sliding window (denoted as IF-s).

2) Evaluation Metrics.: Standard metrics are used to assess downstream task performance. Accuracy (Acc) [65] is applied for classification tasks. Clustering outcomes are measured using Rand Index (RI) and Normalized Mutual Information (NMI) [76], [77]. Anomaly detection employs F1-score [62].

We quantified performance improvements using Cohen's d, computing average effect sizes by comparing our method's gains over baselines across datasets. The metric aggregates normalized differences relative to the pooled standard deviation of all methods' performances, demonstrating our algorithm's consistent superiority.

B. Implementation Details.

The **MMFA** framework is implemented using PyTorch 1.10.2, and all experiments run on a Ubuntu machine equipped with Tesla A100 GPUs. Data augmentation methods are implemented using tsaug [79] with default parameters. Finally, we report the performance after 100 epochs of training.

The majority of **MMFA**'s hyperparameters are consistently assigned fixed values across all experiments, devoid of any hyperparameter optimization. The coefficients α and β in Eq. 12 are both assigned a value of 25, while γ is set to 1. The SGD optimizer is employed to train all feature encoders with a learning rate fixed at 10^{-4}.

The classification baselines and task-specific clustering baselines' outcomes are extracted from the cited publications [9], [65], [68]–[70], [76]. Our reproduced results cover other aspects.

Specifically, we use a window size of 100 and a representation dimension of 32 for the anomaly detection dataset. In the meanwhile we follow the protocols introduced in [64] to evaluate accuracies and F1 scores for the UCR anomaly datasets, except for using isolation forest to produce multiple detections in each sub-dataset to fit with applicable usage.

The specific datasets utilized for each task are outlined below. The original datasets' dimensionalities (channel numbers) can overwhelm 2D transforms (e.g., RP, CWT) and ResNet encoders due to limited computation resources. Therefore, we perform average pooling on transformed multi-channel 2D features, capping dataset channels at 64. Raw time series channels remain unchanged for input into the time series encoder.

VIII. PROOFS OF THEORIES

A. *Theorem 1: Equivalence Between Eigenvalues and Distance Reduction of Spectral Embeddings.*

Proof. We begin by expanding the expected squared difference:

$$\begin{aligned}&\mathbb{E}_{(x,x')\sim p_{sim}}[(f(x)-f(x'))^2]\\&=2\mathbb{E}_{x\sim p'_{data}}[f(x)^2]-2\mathbb{E}_{(x,x')\sim p_{sim}}[f(x)f(x')]\end{aligned}$$

Expand the second term.

$$\begin{aligned}&=2\mathbb{E}_{x\sim p'_{data}}[f(x)^2]\\&\mathbb{E}_{x\sim p'_{data},x'\sim p'_{data}}[f(x)\int_{x'}\frac{\frac{1}{k}p_{T_{(x)},T_{(x')}}(x,x')}{p'_{data}(x)}f(x')dx'] \quad (13)\end{aligned}$$

According to the eigenfunction property.

$$\begin{aligned}&=2\mathbb{E}x\sim p'_{data}[f(x)^2]-\underbrace{2\mathbb{E}x\sim p'_{data}[(1-\lambda)f(x)^2]}_{\text{According to Eq 5.}}\\&=2\lambda\mathbb{E}_{x\sim p'_{data}}[f(x)^2]\end{aligned}$$

Thus, we have established the desired relationship:

$$\mathbb{E}_{(x,x')\sim p_{sim}}[(f(x)-f(x'))^2]=2\lambda\mathbb{E}x\sim p'_{data}[f(x)^2] \quad (14)$$

□

B. *Theorem 2: Multi-modal Invariance Estimation.*

Proof. The proof of this theorem follows from the established inequalities detailed in Inequation 15.

$$\begin{aligned}\mathcal{L}'_{inv}(Z)&=\frac{1}{Nk(k+1)}\sum_{i=1}^{N}(2\sum_{m=1}^{d}||z_i^{(0)}-z_i^{(m)}||_2^2\\&+\sum_{\substack{m=1,n=1\\m\neq n}}^{d}||z_i^{(m)}-z_i^{(n)}||_2^2)\\&\leq\frac{1}{Nk(k+1)}\sum_{i=1}^{N}[2\sum_{m=1}^{d}||z_i^{(0)}-z_i^{(m)}||_2^2\\&+\sum_{\substack{m=1,n=1\\m\neq n}}^{d}\underbrace{(||z_i^{(0)}-z_i^{(m)}||_2^2+||z_i^{(0)}-z_i^{(n)}||_2^2)}_{\textit{Triangle inequation.}}]\\&=\frac{2}{N(k+1)}\sum_{i=1}^{N}\sum_{m=1}^{d}||z_i^{(0)}-z_i^{(m)}||_2^2\end{aligned} \quad (15)$$

□

C. *Theorem 3: Orthogonality of Eigenfunction-Based Representations.*

Proof. When we have the n dimensional matrice $(D-w)$ which has finite rank, $||(D-w)-\mathbb{L}|| \xrightarrow{n\to\infty} 0$ in an approximation sense. Thus, $\mathbb{L}$ is a compact operator. Therefor $\mathbb{L} = \sum_{i=1}^{\infty} \lambda_i \mathbb{E}_{x\sim p'_{data}}[\cdot f_i(x)] f_i(x)$. For all $i \in [d_z]$, we have $\lambda_i \neq 0$, $\mathbb{E}_{x\sim p'_{data}}[f_i(x)^2] = 1$. We prove that a set of d_z eigenfunctions exists that are orthogonal to each other. The operator is symmetric.

$$\mathbb{E}_{x\sim p'_{data}}[\mathbb{L}(f_i)(x)\cdot f_j(x)] = \mathbb{E}_{x\sim p'_{data}}[f_i(x)\cdot \mathbb{L}(f_j)(x)] \quad (16)$$

According to the definition of eigenfunction, we have the equation below.

$$\lambda_i \mathbb{E}_{x\sim p'_{data}}[f_i(x)\cdot f_j(x)] = \lambda_j \mathbb{E}_{x\sim p'_{data}}[f_i(x)\cdot f_j(x)] \quad (17)$$

According to Eq. 17, for any $\lambda_i \neq \lambda_j$ we have:

$$\forall i \neq j, \mathbb{E}_{x\sim p'_{data}}[f_i(x)\cdot f_j(x)] = 0 \quad (18)$$

When we have:

$$\mathcal{S}_\lambda = \{f | \mathbb{L}(f) = \lambda f\} \quad \dim(\mathcal{S}_\lambda) \geq 2 \quad (19)$$

This generates a linear space.

$$\begin{aligned} \mathbb{L}(ah+bg) &= \mathbb{L}(ah) + \mathbb{L}(bg) = \lambda(ah+bg) \\ a,b \in \mathbb{R} \quad h,g &\in \mathcal{S}_\lambda \end{aligned} \quad (20)$$

We have $g_1 \in \mathcal{S}_\lambda$. Then, we remove the subspace spanned by g_1. Now we consider the space $\mathcal{S}^{\dim(\mathcal{S}_\lambda)-1} = \text{span}(\{g_1\})^\perp \cap \mathcal{S}_\lambda$. Then we have $g_i \in \mathcal{S}^{\dim(\mathcal{S}_\lambda)-i+1}$ where $\mathcal{S}^{\dim(\mathcal{S}_\lambda)-i+1} = \text{span}(\{g_j\}_{i-1})^\perp \cap \mathcal{S}_\lambda$. Finally, we have all the $\dim(\mathcal{S}_\lambda)$ orthogonal eigenfunctions with their eigenvalues equal to λ.

□

D. Spectral Selection Mechanism of Linearly Degenerated *MMFA (LD-MMFA)*

This chapter establishes three foundational theorems for **MMFA** under linear degeneracy constraints. We introduce the gradually deduced closed-form solutions for the simple to complex construction of the training objectives.

Introduction of the Unsupervised Objectives. In linearly degenerated **MMFA**, the core objective is to learn a linear projection $Z = XW$ that maps standardized high-dimensional data $X \in \mathbb{R}^{n\times d}$ into a low-dimensional latent space $Z \in \mathbb{R}^{n\times k}$, while preserving critical statistical structures across multiple transformed feature distributions.

The optimization problem is designed jointly.

- (1) **Minimize Projection Complexity**: Penalize the Frobenius norm $\|W\|_F^2$ to select most significant spectrums to enhance generalizability.
- (2) **Align Transformed Features**: Reduce discrepancies $\|XW - F_i W'_i\|_F^2$ between Z and each transformed view $F_i W'_i$.
- (3) **Enforce Orthogonality**: Constrain $Z^T Z = W^\top X^\top X W = I_d$ and $W_i'^\top F_i^\top F_i W'_i = I_d$ to ensure non-redundant, unit-covariance latent features.

We compose these objectives and constraints, deduce their close-form solutions, and find out each spectral selection mechanism below.

Lemma 1. *LD-MMFA Without Transform Alignment Recovers PCA.*

We use $W^\top X^\top X W = I_d$ to forces orthogonality, but minimizing $\|W\|_F^2$ penalizes scaling inversely to variance. Thus, directions with maximal variance (largest λ_i) dominate, as they require minimal scaling to satisfy the orthogonality constraint.

The learning objective of LD-MMFA without transform alignment can be written as:

$$\min_W \mathcal{L} = \|W\|_F^2 \quad s.t. \quad W^\top X^\top X W = I_d. \quad (21)$$

Closed-form solution:

$$W_{opt} = V_d \Lambda_d^{-1/2}, \quad \min_W \mathcal{L} = \sum_{i=1}^{d} \lambda_i^{-1} \quad (22)$$

$\lambda_{X,i}$ is the ith largest singular value of $X^T X$.

Proof. The constrained optimization problem is:

$$\min_W \|W\|_F^2 \quad \text{s.t.} \quad W^\top X^\top X W = I_d. \quad (23)$$

Introduce symmetric Lagrange multiplier matrix Λ:

$$\mathcal{L} = \text{tr}(W^\top W) - \text{tr}\left(\Lambda^\top (W^\top X^\top X W - I_d)\right).$$

Taking derivative w.r.t. W and setting to zero:

$$W = X^\top X W \Lambda \quad \Rightarrow \quad X^\top X W = W (W^\top W)^{-1}.$$

The equation $X^\top X W = W(W^\top W)^{-1}$ implies W's columns span an invariant subspace of $X^\top X$. By the Spectral Theorem, this subspace must be spanned by eigenvectors of $X^\top X$. Let $X^\top X = V \Lambda V^\top$ be the eigenvalue decomposition with $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_p)$ ($\lambda_1 \geq \cdots \geq \lambda_p$).

Express $W = V_d Q$ where V_d contains the top d eigenvectors. The constraint becomes:

$$Q^\top \Lambda_d Q = I_d \quad \Rightarrow \quad Q = \Lambda_d^{-1/2}.$$

Substituting back, $W_{\text{opt}} = V_d \Lambda_d^{-1/2}$. The objective becomes $\sum_{i=1}^{d} \lambda_i^{-1}$, minimized by selecting the largest λ_i (smallest reciprocals).

Thus, the solution recovers PCA's principal directions scaled by $\lambda_i^{-1/2}$, proving equivalence. □

Lemma 2. *LD-MMFA Selects Spectral Consensus with Single Transformed Feature.*

The learning objective of LD-MMFA with single transform alignment can be written as:

$$\begin{aligned} \min_{W,W'} \mathcal{L} &= \|W\|_F^2 + \|W'\|_F^2 + \|XW - FW'\|_F^2 \\ s.t. \quad W^\top X^\top X W &= W'^\top F^\top F W' = I_d. \end{aligned} \quad (24)$$

Closed-Form Solution:

$$W_{opt} = V_X \Lambda_X^{-1/2} U_d, \quad W'_{opt} = V_F \Lambda_F^{-1/2} V_d,$$
$$\min_{W,W'} \mathcal{L} = \sum_{i=1}^{d} \left(\frac{1}{\lambda_{X,i}} + \frac{1}{\lambda_{F,i}} - 2\sigma_i\right) + 2d \tag{25}$$

where U_d, V_d are the top-d singular vectors of $C = V_X \Lambda_X^{-1/2} X^\top F V_F \Lambda_F^{-1/2}$. And $\lambda_{X,i}, \lambda_{F,i}$ and σ_i are the ith largest singular value of $X^T X, F'^\top F'$ and $\bar{X}^T \bar{F}$.

*This optimization problem recovers canonical correlation analysis (CCA) [85] for LD-MMFA and deep/kernel canonical correlation analysis (KCCA) [61] for **MMFA**. This balances regularization penalties while aligning the transformed features.*

Proof. Define whitened matrices $\tilde{X} = X V_X \Lambda_X^{-1/2}$ and $\tilde{F} = F V_F \Lambda_F^{-1/2}$ (satisfying $\tilde{X}^\top \tilde{X} = I$, $\tilde{F}^\top \tilde{F} = I$). Substitute $W = V_X \Lambda_X^{-1/2} Q$, $W' = V_F \Lambda_F^{-1/2} Q'$ to reduce constraints to $Q^\top Q = I_d$, $Q'^\top Q' = I_d$.

The loss becomes:

$$\mathcal{L} = \text{tr}(Q^\top \Lambda_X^{-1} Q) + \text{tr}(Q'^\top \Lambda_F^{-1} Q') - 2\text{tr}(Q^\top C Q') + \text{const.},$$

where $C = \tilde{X}^\top \tilde{F}$ is the whitened cross-covariance matrix.

Maximize $\text{tr}(Q^\top C Q')$ via SVD with $C = U \Sigma V^\top$. Optimal $Q = U_d$, $Q' = V_d$ (top-d singular vectors).

Align W, W' with maximal cross-correlation directions:

$$W_{\text{opt}} = V_X \Lambda_X^{-1/2} U_d, \quad W'_{\text{opt}} = V_F \Lambda_F^{-1/2} V_d.$$

Minimal loss derived as $\sum_{i=1}^{d} (\lambda_{X,i}^{-1} + \lambda_{F,i}^{-1} - 2\sigma_i) + 2d$, where σ_i are top-d singular values of C.

□

Lemma 3. *Multiview Alignment via Joint Cross-Covariance Spectrum. The concatenated matrix C encodes all pairwise correlations between X and each F_i. Its SVD extracts a consensus subspace U_d in X that maximizes the total correlation with all F_i-views.*

Optimization Problem:

$$\min_{W,\{W'_i\}} \mathcal{L} = \|W\|_F^2 + \sum_{i=1}^{p} \left(\|W'_i\|_F^2 + \|XW - F_i W'_i\|_F^2\right)$$
$$\text{s.t.} \quad W^\top X^\top X W = W_i'^\top F_i^\top F_i W'_i = I_d. \tag{26}$$

Closed-Form Solution:

$$W_{opt} = V_X \Lambda_X^{-1/2} U_d \quad W'_{i,opt} = V_{F_i} \Lambda_{F_i}^{-1/2} V_{i,d},$$
$$\min_{W,W'} \mathcal{L} = \sum_{j=1}^{d} \left(\frac{1}{\lambda_{X,j}} + \sum_{i=0}^{p} \frac{1}{\lambda_{F,i,j}} - 2\sigma_j\right) + 2d \tag{27}$$

$\lambda_{X,j}, \lambda_{F,i,j}$ and σ_j are the jth largest singular value of $X^T X, F_i'^\top F'_i$ and $\bar{X}^T \bar{F}$. where U_d and $V_{i,d}$ are derived from the SVD of $C = [\tilde{X}^\top \tilde{F}_1 \ \cdots \ \tilde{X}^\top \tilde{F}_p]$.

Proof. Extending **Lemma** 2 to multiple views, we first whiten X and each F_i:

$$\tilde{X} = X V_X \Lambda_X^{-1/2}, \quad \tilde{F}_i = F_i V_{F_i} \Lambda_{F_i}^{-1/2}. \tag{28}$$

Substituting $W = V_X \Lambda_X^{-1/2} Q$ and $W'_i = V_{F_i} \Lambda_{F_i}^{-1/2} Q'_i$, the constraints reduce to $Q^\top Q = I_d$ and $Q_i'^\top Q'_i = I_d$.

The objective simplifies to maximizing $\sum_{i=1}^{p} \text{tr}(Q^\top \tilde{X}^\top \tilde{F}_i Q'_i)$, which aggregates cross-view correlations. This is equivalent to maximizing $\text{tr}(Q^\top C [Q_1'^\top \ \cdots \ Q_p'^\top]^\top)$, where $C = [\tilde{X}^\top \tilde{F}_1 \ \cdots \ \tilde{X}^\top \tilde{F}_p]$.

Applying SVD to C, we write $C = U \Sigma [V_1^\top \ \cdots \ V_p^\top]^\top$, where U and each V_i are orthogonal. The optimal Q is the first d columns of U, while each Q'_i corresponds to the first d columns of V_i. The minimal loss can be derived with the result.

□

*E. Theorem 4: **MMFA**'s recovery of KPCA and KCCA.*

The following proof demonstrates that under linear degeneracy constraints, **MMFA** recovers Kernel Principal Component Analysis (KPCA) [60] and Kernel Canonical Correlation Analysis (KCCA) [61] when applied to kernel-mapped data. The linear solutions derived in **Lemmas** 1–3 generalize to their kernelized counterparts through the kernel trick, where input data X and transformed features F_i are implicitly mapped into reproducing kernel Hilbert spaces (RKHS).

Proof. When $p = 0$ (no transformed features), we apply LD-MMFA to kernel-mapped data $\Phi_X(X)W$ with $p = 0$. According to **Lemma** 1, the solution to LD-MMFA reduces to KPCA,

When aligning a single transformed feature $\Psi_F(F)W'$ (kernel-mapped from another view F), we apply LD-MMFA to kernel-mapped data $\Phi_X(X)W$ and $\Phi_F(F)W'$ with $p = 1$. According to **Lemma** 2 the solution recovers KCCA, selecting directions of maximal correlation between kernel spaces.

For multiple transformed features $\{\Psi_{F,i}(F_i)W_{F,i}\}$, the solution aligns a consensus subspace across all kernel-induced views, according to **Lemma** 3 generalizing multi-view KCCA. □

IX. Conclusion

In this study, we propose a framework aimed at mitigating the limitations arising from intrinsic feature engineering in Universal Representation Learning (URL) for Multivariate Time Series (MTS). The incorporation of feature alignment and regularization methods forms a revisitation of the spectral selection mechanism for constructing URL framework, surpassing existing state-of-the-art approaches.

Looking ahead, this study unveils several promising avenues. For instance, delving deeper into transform property deduction and more efficient learning methods could yield more comprehensive representations. The integration of broader external knowledge or domain-specific information holds the potential to further enrich these representations and accelerate the learning process.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (NSFC) (72371084)

REFERENCES

[1] S. Bennett, M. Cucuringu, and G. Reinert, "Detection and clustering of lead-lag networks for multivariate time series with an application to financial markets," 2022.

[2] E. Kim, S. Cho, B. Lee, and M. Cho, "Fault detection and diagnosis using self-attentive convolutional neural networks for variable-length sensor data in semiconductor manufacturing," *IEEE Transactions on Semiconductor Manufacturing*, vol. 32, no. 3, pp. 302–309, 2019.

[3] A. Riley and E. Nica, "Internet of things-based smart healthcare systems and wireless biomedical sensing devices in monitoring, detection, and prevention of covid-19." *American Journal of Medical Research*, vol. 8, no. 2, 2021.

[4] K. Bi, L. Xie, H. Zhang, X. Chen, X. Gu, and Q. Tian, "Accurate medium-range global weather forecasting with 3d neural networks," *Nature*, pp. 1–6, 2023.

[5] S. Tonekaboni, D. Eytan, and A. Goldenberg, "Unsupervised representation learning for time series with temporal neighborhood coding," in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. [Online]. Available: https://openreview.net/forum?id=8qDwejCuCN

[6] E. Eldele, M. Ragab, Z. Chen, M. Wu, C. K. Kwoh, X. Li, and C. Guan, "Time-series representation learning via temporal and contextual contrasting," in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, Z. Zhou, Ed. ijcai.org, 2021, pp. 2352–2359. [Online]. Available: https://doi.org/10.24963/ijcai.2021/324

[7] L. Yang and S. Hong, "Unsupervised time-series representation learning with iterative bilinear temporal-spectral fusion," in *International Conference on Machine Learning*. PMLR, 2022, pp. 25 038–25 054.

[8] Z. Liang, J. Zhang, C. Liang, H. Wang, Z. Liang, and L. Pan, "A shapelet-based framework for unsupervised multivariate time series representation learning," *Proceedings of the VLDB Endowment*, vol. 17, no. 3, pp. 386–399, 2023.

[9] Z. Yue, Y. Wang, J. Duan, T. Yang, C. Huang, Y. Tong, and B. Xu, "Ts2vec: Towards universal representation of time series," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 8, 2022, pp. 8980–8987.

[10] G. Zerveas, S. Jayaraman, D. Patel, A. Bhamidipaty, and C. Eickhoff, "A transformer-based framework for multivariate time series representation learning," in *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 2021, pp. 2114–2124.

[11] J. Paparrizos and S. P. T. Reddy, "Odyssey: An engine enabling the time-series clustering journey," *Proceedings of the VLDB Endowment*, vol. 16, no. 12, pp. 4066–4069, 2023.

[12] A. Bonifati, F. Del Buono, F. Guerra, and D. Tiano, "Time2feat: learning interpretable representations for multivariate time series clustering," *Proceedings of the VLDB Endowment (PVLDB)*, vol. 16, no. 2, pp. 193–201, 2022.

[13] S. Schmidl, P. Wenig, and T. Papenbrock, "Anomaly detection in time series: a comprehensive evaluation," *Proceedings of the VLDB Endowment*, vol. 15, no. 9, pp. 1779–1797, 2022.

[14] D. Zhu, D. Song, Y. Chen, C. Lumezanu, W. Cheng, B. Zong, J. Ni, T. Mizoguchi, T. Yang, and H. Chen, "Deep unsupervised binary coding networks for multivariate time series retrieval," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 02, 2020, pp. 1403–1411.

[15] J. Paparrizos and M. J. Franklin, "Grail: efficient time-series representation learning," *Proceedings of the VLDB Endowment*, vol. 12, no. 11, pp. 1762–1777, 2019.

[16] X. Guo, L. Gao, X. Liu, and J. Yin, "Improved deep embedded clustering with local structure preservation." in *Ijcai*, vol. 17, 2017, pp. 1753–1759.

[17] M. Balasubramanian and E. L. Schwartz, "The isomap algorithm and topological stability," *Science*, vol. 295, no. 5552, pp. 7–7, 2002.

[18] X. Pan, L. Hou, M. Stephen, H. Yang, and C. Zhu, "Evaluation of scaling invariance embedded in short time series," *PLoS One*, vol. 9, no. 12, p. e116128, 2014.

[19] X. He and P. Niyogi, "Locality preserving projections," *Advances in neural information processing systems*, vol. 16, 2003.

[20] G. E. Batista, X. Wang, and E. J. Keogh, "A complexity-invariant distance measure for time series," in *Proceedings of the 2011 SIAM international conference on data mining*. SIAM, 2011, pp. 699–710.

[21] Z. Wang, T. Oates *et al.*, "Encoding time series as images for visual inspection and classification using tiled convolutional neural networks," in *Workshops at the twenty-ninth AAAI conference on artificial intelligence*, vol. 1. AAAI Menlo Park, CA, USA, 2015.

[22] S.-H. Park, N. S. Syazwany, and S.-C. Lee, "Meta-feature fusion for few-shot time series classification," *IEEE Access*, 2023.

[23] P. Schäfer and M. Högqvist, "Sfa: a symbolic fourier approximation and index for similarity search in high dimensional datasets," in *Proceedings of the 15th international conference on extending database technology*, 2012, pp. 516–527.

[24] W. Tang, L. Liu, and G. Long, "Interpretable time-series classification on few-shot samples," in *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020, pp. 1–8.

[25] E. Abbe, S. Bengio, A. Lotfi, and K. Rizk, "Generalization on the unseen, logic reasoning and degree curriculum," in *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, ser. Proceedings of Machine Learning Research, A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, Eds., vol. 202. PMLR, 2023, pp. 31–60. [Online]. Available: https://proceedings.mlr.press/v202/abbe23a.html

[26] S. Ren, Z. Gao, T. Hua, Z. Xue, Y. Tian, S. He, and H. Zhao, "Co-advise: Cross inductive bias distillation," in *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, 2022, pp. 16 773–16 782.

[27] J. Robinson, L. Sun, K. Yu, K. Batmanghelich, S. Jegelka, and S. Sra, "Can contrastive learning avoid shortcut solutions?" *Advances in neural information processing systems*, vol. 34, pp. 4974–4986, 2021.

[28] L. Le Cam and G. L. Yang, "On the preservation of local asymptotic normality under information loss," *The Annals of Statistics*, pp. 483–520, 1988.

[29] R. Geirhos, J.-H. Jacobsen, C. Michaelis, R. Zemel, W. Brendel, M. Bethge, and F. A. Wichmann, "Shortcut learning in deep neural networks," *Nature Machine Intelligence*, vol. 2, no. 11, pp. 665–673, 2020.

[30] A. M. Puli, L. Zhang, Y. Wald, and R. Ranganath, "Don't blame dataset shift! shortcut learning due to gradients and cross entropy," *Advances in Neural Information Processing Systems*, vol. 36, pp. 71 874–71 910, 2023.

[31] A. Grossmann, R. Kronland-Martinet, and J. Morlet, "Reading and understanding continuous wavelet transforms," in *Wavelets: Time-Frequency Methods and Phase Space Proceedings of the International Conference, Marseille, France, December 14–18, 1987*. Springer, 1990, pp. 2–20.

[32] Q. Meng, H. Qian, Y. Liu, Y. Xu, Z. Shen, and L. Cui, "Unsupervised representation learning for time series: A review," *arXiv preprint arXiv:2308.01578*, 2023.

[33] P. Vadiraja and M. A. Chattha, "A survey on knowledge integration techniques with artificial neural networks for seq-2-seq/time series models," *arXiv preprint arXiv:2008.05972*, 2020.

[34] Z. Li, Z. Rao, L. Pan, P. Wang, and Z. Xu, "Ti-mae: Self-supervised masked time series autoencoders," *CoRR*, vol. abs/2301.08871, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2301.08871

[35] R. Liu, W. Ma, E. L. Zippi, H. Pouransari, J. Xiao, C. M. Sandino, B. Mahasseni, J. Minxha, E. Azemi, E. L. Dyer, and A. Moin, "Generalizable autoregressive modeling of time series through functional narratives," *CoRR*, vol. abs/2410.08421, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2410.08421

[36] Q. Ma, J. Zheng, S. Li, and G. W. Cottrell, "Learning representations for time series clustering," *Advances in neural information processing systems*, vol. 32, 2019.

[37] P. Malhotra, V. TV, L. Vig, P. Agarwal, and G. Shroff, "Timenet: Pretrained deep recurrent neural network for time series classification," in *25th European Symposium on Artificial Neural Networks, ESANN 2017, Bruges, Belgium, April 26-28, 2017*, 2017. [Online]. Available: https://www.esann.org/sites/default/files/proceedings/legacy/es2017-100.pdf

[38] T. Huynh, S. Kornblith, M. R. Walter, M. Maire, and M. Khademi, "Boosting contrastive self-supervised learning with false negative cancellation," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2022, pp. 2785–2795.

[39] J.-Y. Franceschi, A. Dieuleveut, and M. Jaggi, "Unsupervised scalable representation learning for multivariate time series," *Advances in neural information processing systems*, vol. 32, 2019.

[40] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "Timesnet: Temporal 2d-variation modeling for general time series analysis," in *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. [Online]. Available: https://openreview.net/forum?id=ju_Uqw384Oq

[41] Z. Li, Z. Rao, L. Pan, P. Wang, and Z. Xu, "Ti-mae: Self-supervised masked time series autoencoders," *arXiv preprint arXiv:2301.08871*, 2023.

[42] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. E. Hinton, "Big self-supervised models are strong semi-supervised learners," *Advances in neural information processing systems*, vol. 33, pp. 22 243–22 255, 2020.

[43] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[44] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[45] J. Li, P. Zhou, C. Xiong, and S. C. H. Hoi, "Prototypical contrastive learning of unsupervised representations," in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. [Online]. Available: https://openreview.net/forum?id=KmykpuSrjcq

[46] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," *Advances in neural information processing systems*, vol. 33, pp. 9912–9924, 2020.

[47] Q. Meng, H. Qian, Y. Liu, L. Cui, Y. Xu, and Z. Shen, "Mhccl: Masked hierarchical cluster-wise contrastive learning for multivariate time series," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 8, 2023, pp. 9153–9161.

[48] A. Hyvarinen and H. Morioka, "Unsupervised feature extraction by time-contrastive learning and nonlinear ica," *Advances in neural information processing systems*, vol. 29, 2016.

[49] S. Winograd, "On computing the discrete fourier transform," *Mathematics of computation*, vol. 32, no. 141, pp. 175–199, 1978.

[50] A. Notaristefano, G. Chicco, and F. Piglione, "Data size reduction with symbolic aggregate approximation for electrical load pattern grouping," *IET Generation, Transmission & Distribution*, vol. 7, no. 2, pp. 108–117, 2013.

[51] M. Wolter, F. Blanke, J. Garcke, and C. T. Hoyt, "ptwt - the pytorch wavelet toolbox," *Journal of Machine Learning Research*, vol. 25, no. 80, pp. 1–7, 2024. [Online]. Available: http://jmlr.org/papers/v25/23-0636.html

[52] A. Gupta and V. Kumar, "The scalability of fft on parallel computers," *IEEE Transactions on Parallel and Distributed Systems*, vol. 4, no. 8, pp. 922–932, 2002.

[53] R. Girdhar, A. El-Nouby, Z. Liu, M. Singh, K. V. Alwala, A. Joulin, and I. Misra, "Imagebind: One embedding space to bind them all," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 15 180–15 190.

[54] G. Yariv, I. Gat, L. Wolf, Y. Adi, and I. Schwartz, "Audiotoken: Adaptation of text-conditioned diffusion models for audio-to-image generation," *arXiv preprint arXiv:2305.13050*, 2023.

[55] A. Bardes, J. Ponce, and Y. LeCun, "Vicreg: Variance-invariance-covariance regularization for self-supervised learning," in *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. [Online]. Available: https://openreview.net/forum?id=xm6YD62D1Ub

[56] R. Balestriero and Y. LeCun, "Contrastive and non-contrastive self-supervised learning recover global and local spectral embedding methods," *Advances in Neural Information Processing Systems*, vol. 35, pp. 26 671–26 685, 2022.

[57] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[58] N. G. Trillos, F. Hoffmann, and B. Hosseini, "Geometric structure of graph laplacian embeddings," *Journal of Machine Learning Research*, vol. 22, no. 63, pp. 1–55, 2021.

[59] L. Trevisan, "Lecture notes on graph partitioning, expanders and spectral methods," *University of California, Berkeley, https://people. eecs. berkeley. edu/luca/books/expanders-2016. pdf*, 2017.

[60] B. Schölkopf, A. J. Smola, and K. Müller, "Kernel principal component analysis," in *Artificial Neural Networks - ICANN '97, 7th International Conference, Lausanne, Switzerland, October 8-10, 1997, Proceedings*, ser. Lecture Notes in Computer Science, W. Gerstner, A. Germond, M. Hasler, and J. Nicoud, Eds., vol. 1327. Springer, 1997, pp. 583–588. [Online]. Available: https://doi.org/10.1007/BFb0020217

[61] K. Fukumizu, F. R. Bach, and A. Gretton, "Statistical consistency of kernel canonical correlation analysis." *Journal of Machine Learning Research*, vol. 8, no. 2, 2007.

[62] Z. Li, Y. Zhao, J. Han, Y. Su, R. Jiao, X. Wen, and D. Pei, "Multivariate time series anomaly detection and interpretation using hierarchical inter-metric and temporal embedding," in *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 2021, pp. 3220–3230.

[63] Y. Su, Y. Zhao, C. Niu, R. Liu, W. Sun, and D. Pei, "Robust anomaly detection for multivariate time series through stochastic recurrent neural network," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 2828–2837.

[64] R. Wu and E. J. Keogh, "Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, pp. 2421–2429, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:221995939

[65] 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.

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

[67] X. Zhang, Y. Gao, J. Lin, and C.-T. Lu, "Tapnet: Multivariate time series classification with attentional prototypical network," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, 2020, pp. 6845–6852.

[68] G. Li, B. Choi, J. Xu, S. S. Bhowmick, K.-P. Chun, and G. L.-H. Wong, "Shapenet: A shapelet-neural network approach for multivariate time series classification," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 9, 2021, pp. 8375–8383.

[69] 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 *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. [Online]. Available: https://openreview.net/forum?id=PDYs7Z2XFGv

[70] Q. Xiao, B. Wu, Y. Zhang, S. Liu, M. Pechenizkiy, E. Mocanu, and D. C. Mocanu, "Dynamic sparse network for time series classification: Learning what to "see"," *Advances in Neural Information Processing Systems*, vol. 35, pp. 16 849–16 862, 2022.

[71] J. Lines, S. Taylor, and A. Bagnall, "Time series classification with hive-cote: The hierarchical vote collective of transformation-based ensembles," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 12, no. 5, pp. 1–35, 2018.

[72] H. Li, "Multivariate time series clustering based on common principal component analysis," *Neurocomputing*, vol. 349, pp. 239–247, 2019.

[73] K. Ø. Mikalsen, F. M. Bianchi, C. Soguero-Ruiz, and R. Jenssen, "Time series cluster kernel for learning similarities between multivariate time series with missing data," *Pattern Recognition*, vol. 76, pp. 569–581, 2018.

[74] M. Ozer, A. Sapienza, A. Abeliuk, G. Muric, and E. Ferrara, "Discovering patterns of online popularity from time series," *Expert Systems with Applications*, vol. 151, p. 113337, 2020.

[75] D. Ienco and R. Interdonato, "Deep multivariate time series embedding clustering via attentive-gated autoencoder," in *Advances in Knowledge Discovery and Data Mining: 24th Pacific-Asia Conference, PAKDD 2020, Singapore, May 11–14, 2020, Proceedings, Part I 24*. Springer, 2020, pp. 318–329.

[76] N. Zhang and S. Sun, "Multiview unsupervised shapelet learning for multivariate time series clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4981–4996, 2022.

[77] Q. Zhang, J. Wu, P. Zhang, G. Long, and C. Zhang, "Salient subsequence learning for time series clustering," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 9, pp. 2193–2207, 2018.

[78] L. A. Becker, "Effect size (es)," 2000.

[79] A. A. Inc. (2019) tsaug: An open-source package for time series data augmentation. [Online]. Available: https://tsaug.readthedocs.io/en/stable/references.html

[80] L. J. Hubert and P. Arabie, "Comparing partitions," *Journal of Classification*, vol. 2, pp. 193–218, 1985.

[81] A. Strehl and J. Ghosh, "Cluster ensembles — a knowledge reuse framework for combining multiple partitions," *J. Mach. Learn. Res.*, vol. 3, pp. 583–617, 2002.

[82] C. Cortes and V. N. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, pp. 273–297, 1995.

[83] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," *2008 Eighth IEEE International Conference on Data Mining*, pp. 413–422, 2008.

[84] J. MacQueen, "Some methods for classification and analysis of multivariate observations," 1967.

[85] B. Thompson, "Canonical correlation analysis." 2000.