

Received 21 September 2025, accepted 16 October 2025, date of publication 20 October 2025, date of current version 4 November 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3623930

RESEARCH ARTICLE

Re-Ranking and Representations for Time Series Retrieval: A Comparative Study

BIONDA ROZIN¹, DANIEL CARLOS GUIMARÃES PEDRONETTE¹, AND RICARDO DA SILVA TORRES², (Member, IEEE)

¹Department of Statistics, Applied Mathematics, and Computing (DEMAC), Universidade Estadual Paulista Júlio De Mesquita Filho (UNESP), Rio Claro 04266-030, Brazil

²Artificial Intelligence Group, Wageningen University and Research (WUR), 6708 PB Wageningen, The Netherlands

Corresponding author: Daniel Carlos Guimarães Pedronette (daniel.pedronette@unesp.br)

This work was supported in part by São Paulo Research Foundation-FAPESP under Grant 2022/01359-1, Grant 2023/08087-0, and Grant 2018/15597-6; in part by Brazilian National Council for Scientific and Technological Development-CNPq under Grant 313193/2023-1 and Grant 422667/2021-8; in part by Coordenação de Aperfeiçoamento de Pessoal de Nível Superior-Brasil (CAPES) Article Processing Charge (APC) was covered by CAPES under Grant ROR ID: 00x0ma614.

ABSTRACT Properly understanding the trends and patterns of multiple variables over time is important for decision-making in several applications, ranging from ecological monitoring based on vegetation index time series to health condition assessment based on physiological indicator temporal profiles. In this context, effective time series retrieval systems, which involve time series pattern representations and similarity ranking, are vital. This paper presents a comparative study that includes different techniques for time series representations and ranking. Special attention is given to the comparison of the retrieval effectiveness performance of re-ranking methods based on unsupervised distance learning for time series. Conducted experiments included the comparison of 10 distinct representations and four different re-ranking approaches on six diverse time series datasets. Experimental results demonstrate that the proper combination of representation and re-ranking methods can often enhance the overall quality of ranking results, leading to gains on mAP up to 31.78%. Also, the proposed approach is flexible, allowing the use of other feature extractors, distance measures, and re-ranking methods. The results suggest that the choice of time series representation and re-ranking method depends on the specific application context.

INDEX TERMS Time series retrieval, time series representation, re-ranking, unsupervised distance learning, comparative study.

I. INTRODUCTION

The widespread use of sensing technologies, as well as the reduction in costs related to data storage, led to the creation of huge collections of time series. Identifying trends and patterns associated with specific variables over time is critical for proper decision-making. Promising applications of time series analysis tools include finances [1], machinery maintenance [2], weather monitoring [3], outlier detection [4], phenology studies [5], dispersion of volcanic plume height [6], EEG signal analysis [7], estimation of sales of products and services during a time period [8], among others.

The associate editor coordinating the review of this manuscript and approving it for publication was Donato Impedovo¹.

The creation of efficient and effective time series search systems is of paramount importance. In particular, similarity search systems, in which collection time series are ranked according to their similarity to a given query (e.g., input time series), have been explored to provide users with means to identify time series of interest [5], [6], [7], [9], [10], [11], [12], [13], [14], [15], [16], [17].

The assessment of the similarity of two time series often relies on the extraction of representations, commonly known as feature vectors, which are expected to encode relevant patterns of time series [18]. In this time series retrieval formulation, the similarity calculation is seen as a function of the proximity of those vectors in the feature space. The closer the two vectors are, the more similar the time series are expected to be. Different approaches have

been proposed to extract meaningful representations from time series [19], [20], [21], [22], [23]. Information, such as shape [24], temporal correlation [19], and time series trajectory [25], has also been considered to obtain time series representations. The most recent retrieval approaches for time series employ deep learning models [7], [9], [10], [16], [17], which have led to great improvements in results. However, they face challenges related to the requirement for large labeled datasets to train deep networks and the associated computational costs. In this direction, time series retrieval strategies that are effective, fast, and computationally inexpensive are still highly necessary in practice. In this context, therefore, the identification of suitable time series representations for retrieval tasks is still an open problem in the literature.

The effectiveness of time series retrieval tasks is influenced not only by the representation but also by the distance (or similarity) functions employed to assess how close two feature vectors are. The use of reliable and effective distance measurements between the data objects is of fundamental importance. Generally, the distance measures used, such as the Euclidean distance or the Dynamic Time Warping [26], consider only pairwise relationships [12]. The Euclidean Distance is often employed in scenarios of normalized time series comparison, avoiding effects of scale, offset, and distortion [27]. DTW, on the other hand, is an elastic distance measurement, often employed for time series alignment, being useful in applications of student performance prediction based on behavior data, for example [28].

However, some limitations are observed. In [29], we observe inferior performance of DTW in clustering tasks in comparison to the Euclidean Distance, regarding normalization, where the choice of the latter method affects the Euclidean Distance performance. Also, pairwise relationships often despise the existing contextual information in the whole dataset. In this context, rank-based unsupervised distance learning algorithms are a powerful tool, capable of analyzing the entire dataset structure and considering neighbor relationships to compute more effective distance measures. Unsupervised and semi-supervised methods are promising as they do not require large volumes of labeled data, which can be expensive and time-consuming to obtain [30]. Different approaches have recently been proposed in the literature. One research line refers to the use of graphs and hypergraphs, which are utilized to analyze connections between different points and discover new relationships in a dataset [31], [32]. These methods were extensively evaluated on image datasets, achieving competitive results compared to other ranking algorithms. Valem et al. [33], [34], in turn, considered re-ranking formulations based on hypergraph approaches. Their methodology for person re-ID achieved competitive results when compared with state-of-the-art re-identification methods. In [35], Sarfraz et al. proposed a re-id method based on an embedding scheme. Their method also integrated a re-ranking method based on expanded cross neighborhoods.

The investigation of the use of diffusion processes also represents a relevant research direction. In [36], a diffusion process is performed for similarity fusion in retrieval tasks. The target application was 3D shape retrieval. Iscen et al. [37] introduced a method based on metric learning, considering Euclidean distances and manifold learning distances. The results achieved were on par with or outperformed the prior fully or partially supervised models. Manifold learning algorithms can be used along with other machine learning tasks to improve results. Chen et al. [9], for example, proposed a deep multi-task representation learning method (MTRL) for time series classification and retrieval tasks, leveraging supervised and unsupervised information. Supervised information for classification tasks aims to reduce intra-class differences and increase inter-class differences. Unsupervised information for retrieval tasks aims to preserve Dynamic Time Warping (DTW) distances between pairs. These tasks share networks, so the information obtained from one task can benefit the other. Almeida et al. [38] investigated the use of a re-ranking method in the context of time series retrieval. In their study, time series were obtained from images captured by cameras positioned in towers within vegetation areas. Retrieval of time series is conducted using a re-ranking approach, which is exploited to support the analysis of vegetation phenological changes. Their study has not considered time series representations nor multiple re-ranking methods. The work in [23] employs eight methods for time series retrieval, including traditional approaches, neural networks, and a Residual Network 2D with Template Learning, proposed by the authors, outperforming the other reported methods. Representation and contextual information are not explored in this work. To the best of our knowledge, the investigation of the potential of the time series representation in conjunction with re-ranking methods based on unsupervised learning for time series retrieval tasks is a problem overlooked in the literature.

This paper presents a comparative study that includes different methods for effective time series retrieval. We compare diverse feature extractors for time series representation and assess their ability to encode relevant time series patterns. An initial ranking step is performed utilizing the Euclidean distance, and a re-ranking step is performed utilizing diverse unsupervised algorithms, to explore the whole data structure and obtain more effective distance measures. Also, the proposed pipeline is flexible, allowing the use of different representations, distance measurements, and re-ranking approaches. The main objective of our comparative study is to explore the ability of unsupervised distance learning methods to improve time series retrieval results, considering different datasets and representation methods. In summary, our main contributions are:

- Comparative study involving ten different time series representation methods in time series retrieval tasks;
- Comparative study involving four different re-ranking methods based on unsupervised distance learning algorithms in time series retrieval tasks;

- Application of a flexible pipeline for time series retrieval, based on content-based image retrieval [39], using representation approaches and unsupervised contextual similarity measurements;
- Comparative study of different retrieval systems created based on the combination of different time series representations and unsupervised distance learning approaches. To the best of our knowledge, this work is the first initiative to consider such a scenario. Although related work [38] has previously employed an unsupervised distance learning method, representation approaches were not jointly exploited.

The rest of the paper is organized as follows: Section II presents a formal definition of the problem, Section III details the time series description and re-ranking methods included in the comparative study. Section IV presents the experimental evaluation and Section V presents the results, while Section VI presents our conclusions and points out directions for future work.

II. TIME SERIES RETRIEVAL MODEL

This section presents the time series retrieval model employed in this paper. The model incorporates both time series representation and re-ranking methods based on unsupervised distance learning.

We follow the retrieval models proposed for image search [39] and re-ranking [40], [41]. Let $\mathcal{C} = \{T_1, T_2, \dots, T_n\}$ be a collection composed of n time series. Let D be a time series descriptor, which can be defined by the tuple (ϵ, ρ) , where $\epsilon : T_i \rightarrow \mathbb{R}^d$ is a function that extracts a feature vector v_{T_i} from a time series T_i , and $\rho : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}_0^+$ is a function that computes the distance between two time series T_i and T_j based on their corresponding feature vectors, formally defined as $\rho(\epsilon(T_i), \epsilon(T_j))$ or simply $\rho(i, j)$.

The distance $\rho(i, j)$ for each pair (i, j) can be computed to form a square matrix A of dimension $n \times n$, where $A_{ij} = \rho(i, j)$. Based on the distance ρ , a ranked list τ_q is computed for each time series T_q present in the collection. The ranked list $\tau_q = \langle T_1, T_2, \dots, T_n \rangle$ is a permutation of the collection \mathcal{C} , where $\tau_q(i)$ is the rank of the time series T_i in τ_q . The rank is defined according to the distance measure, such that if $\tau_q(i) < \tau_q(j)$, then $\rho(q, i) < \rho(q, j)$. The set of ranked lists for each time series in \mathcal{C} is defined as $\mathcal{R} = \{\tau_1, \tau_2, \dots, \tau_n\}$.

The re-ranking task can be seen as recalculating the distance ρ using a more effective distance function ρ_r . The function $\rho_r(i, j, \mathcal{R})$ aims to explore the contextual information present in the set \mathcal{R} , analyzing the relationships existing in a k -neighborhood of the ranked lists and improving the effectiveness of distances between time series. Formally, $\rho_r : \mathcal{C} \times \mathcal{C} \times \mathcal{R} \rightarrow \mathbb{R}_0^+$ is a distance function between time series T_i and T_j that takes into account the contextual similarity information contained in the set \mathcal{R} .

III. COMPARATIVE STUDY

This paper presents and discusses a comparative study that assesses different time series representations and re-ranking

algorithms. Section III-A discusses a general perspective of the study, while Sections III-B and III-C present the representation and re-ranking approaches considered.

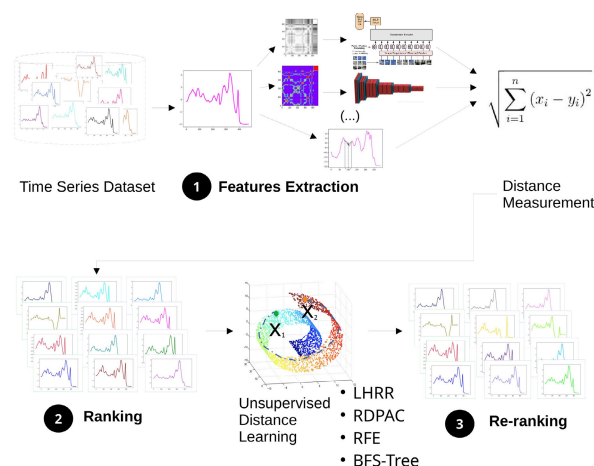


FIGURE 1. Pipeline employed in the comparative study.

TABLE 1. Summary of the utilized descriptors.

Descriptor	Type
Beam Angle Statistics (BAS)	Shape
Discrete Fourier Transform (DFT)	Frequency
Discrete Wavelet Transform (DWT)	Frequency
Random Convolutional Kernel Transform (ROCKET)	Convolution
MOMENT	Transformers
Gramian Angular Field (GAF) + ResNet-152	Image-based
Markov Transition Field (MTF) + ResNet-152	Image-based
Recurrence Plot (RP) + ResNet-152	Image-based
Gramian Angular Field (GAF) + ViT	Image-based
Markov Transition Field (MTF) + ViT	Image-based
Recurrence Plot (RP) + ViT	Image-based

A. OVERVIEW

Fig. 1 presents a schematic view of the procedures conducted in the comparative study. First, time series are represented through different feature extraction strategies, including methods that handle time series as a 1D signal and those that convert time series into 2D representations that are then combined with deep-learning-based feature extractors (Section III-B). For different queries, the collection time series are ranked according to the distance of their representations to the queries. This ranking procedure is based on the Euclidean distance. Later, re-ranking methods (Section III-C), based on unsupervised distance learning algorithms, are employed. The goal is not only to derive the most promising representations and re-ranking methods for different time series retrieval scenarios and identify the combinations that would lead to effective results, but also to employ a flexible pipeline for time series, where other representation methods, distance measurements, and unsupervised distance learning approaches can be applied, and ranking results would also be satisfactory.

B. TIME SERIES REPRESENTATIONS

Diverse approaches for time series representation were considered in our study. Tab. 1 presents a summary of descriptors employed. The next sections introduce each approach.

1) 1D-SIGNAL-BASED FEATURE EXTRACTORS

Effective machine learning relies on robust data representation, particularly for time series. Feature extraction from time series is pivotal for accurate information retrieval, encompassing various properties, such as shapes, frequencies, pattern repetitions, and temporal correlations.

a: BEAM ANGLE STATISTIC

The Beam Angle Statistics (BAS) [24], [42] is a shape descriptor that relies on identifying concavities and convexities in contours and extracting related features. The method, including in prior research [38], [43], has demonstrated promise [38]. By considering a parameter k_b , a contour described by points $P = (p_1, p_2, \dots, p_n)$ undergoes traversal. Features are generated based on the angle α_i , formed by the intersection of line segments x (connecting points $(i - k_b, p_{i-k_b})$ and (i, p_i)) and y (connecting points (i, p_i) and $(i + k_b, p_{i+k_b})$), where $i = k_b, k_b + 1, k_b + 2, \dots, n - k_b$. This yields the representation $A = (\alpha_1, \alpha_2, \dots, \alpha_{n-2k_b})$ for any contour.

b: DISCRETE FOURIER TRANSFORM (DFT)

The Discrete Fourier Transform (DFT) preserves the point count while transforming a signal. It dissects time series into basis functions, where early functions capture trends and later ones account for noise. Each function possesses a complex Fourier coefficient. The initial coefficients approximate the time series [44].

c: DISCRETE WAVELET TRANSFORM (DWT)

The Discrete Wavelet Transform (DWT) breaks time series into diverse frequencies across scales, reducing dimensions and/or noise [45]. It yields coarse-grained approximation and fine-grained detail coefficients through specific wavelet functions, involving convolution and downsampling.

d: RANDOM CONVOLUTIONAL KERNEL TRANSFORM (ROCKET)

ROCKET [46] employs kernels for time series feature extraction. It generates k_r convolutional kernels, extracting maximum values and positive proportion per convolution. ROCKET excels in efficiency, boasting lower computational complexity than alternatives, needing just one hyperparameter evaluation, the number of kernels k_r .

e: MOMENT

MOMENT [47] is a family of pre-trained transformer-based models for time series analysis. The proposed foundation model is pre-trained on the *Time Series Pile*, a repository

comprising time series from multiple domains and characteristics. The model first breaks down time series into patches and learns robust representations through a masked modeling objective, reconstructing randomly masked patches from the large, unlabeled dataset. The MOMENT is useful to generate robust and generalizable features for a wide range of machine learning tasks.

The descriptors were selected based on many criteria. DWT and DFT were chosen as representatives of traditional approaches, i.e., methods already well-established in the field [48]. DFT is a descriptor robust to noise that reduces dimensionality while preserving global information. DWT, on the other hand, details both global (Approximation) and local (Detail) information, finding patterns and also being robust to noise. ROCKET is a recent algorithm that can be used for time series classification and also feature extraction, and, to the best of our knowledge, has not been applied in information retrieval tasks. ROCKET employs diverse random convolution filters over time series, exploring diverse characteristics with efficiency, scalability, and strong generalization of features. The use of the BAS algorithm is based on [38]. BAS is a representative of the family of methods that rely on shape-based characterization. This representation allows focusing on angles, making the features invariant to eventual amplitude or offsets of different time series. MOMENT was chosen as a new transformer-based methodology for time series analysis, where transfer learning concepts are employed, generating robust features while taking into account both local and global structures of the time series with strong generalization.

2) IMAGE-BASED TIME SERIES DESCRIPTION

The representation of time series through images, which are subsequently exploited for feature extraction, is a promising approach [19], [25], as demonstrated in several applications, ranging from phenology [49] to land cover analysis studies [50], [51], [52]. In those methods, time series patterns are encoded in bi-dimensional representations and pre-trained models (e.g., based on transfer learning procedures relying on Convolutional Neural Networks and Transformers-based models) are used for feature extraction. This approach is capable of exploring diverse color, shape, and texture patterns represented by time series imaging methods, achieving promising outcomes in other machine learning approaches, and holding promise for retrieval tasks. In the following, we discuss the approaches used for obtaining the bi-dimensional representations and the deep learning approaches employed for feature extraction.

a: GRAMIAN ANGULAR FIELDS (GAF)

This method generates an image based on polar coordinates of a time series $T = [t_1, t_2, \dots, t_N]$. It transforms normalized series T into \tilde{T} using polar coordinates in $[-1, 1]$. Using \tilde{T} , a Gramian Angular Field Matrix [19] ($N \times N$) forms, depicted

in Equation 1:

$$GAF = \tilde{T}'\tilde{T} - \sqrt{I - \tilde{T}^2} \sqrt{I - \tilde{T}^2} \quad (1)$$

Here, I is a unit line vector. Each point $GAF_{i,j}$ in the Gramian Angular Field Matrix calculates the trigonometric sum of angles between corresponding points in \tilde{T} , representing the temporal correlation between different time intervals in the series.

b: MARKOV TRANSITION FIELDS (MTF)

This method creates time series images based on Markov transition probabilities. Given a time series $T = [t_1, t_2, \dots, t_N]$, it is identified Q quantile bins and assigned each t_i to the corresponding bin q_j ($j \in [1, Q]$), that divide the sorted series into equal subsets. A $Q \times Q$ weighted adjacency matrix W is constructed by counting transitions among quantile bins, employing a first-order Markov chain methodology along the temporal axis. $w_{i,j}$ represents the frequency at which a point in quantile q_j is followed by a point in quantile q_i . After normalization by $\sum_j w_{i,j} = 1$, we have the Markov Transition Matrix, defined as in Equation 2:

$$W = \begin{bmatrix} w_{11}|P(t_x \in q_1 | t_{x-1} \in q_1) & \dots & w_{1Q}|P(t_x \in q_1 | t_{x-1} \in q_Q) \\ w_{21}|P(t_x \in q_2 | t_{x-1} \in q_1) & \dots & w_{2Q}|P(t_x \in q_2 | t_{x-1} \in q_Q) \\ \dots & \dots & \dots \\ w_{Q1}|P(t_x \in q_Q | t_{x-1} \in q_1) & \dots & w_{QQ}|P(t_x \in q_Q | t_{x-1} \in q_Q) \end{bmatrix} \quad (2)$$

W is insensitive to the distribution of the time series T and temporal dependency on time steps x_i . This results in information loss. Next, a Markov Transition Field [19] of dimensions $N \times N$ is built per Equation 3:

$$MTF = \begin{bmatrix} w_{ij}|t_1 \in q_i, t_1 \in q_j & \dots & w_{ij}|t_1 \in q_i, t_n \in q_j \\ w_{ij}|t_2 \in q_i, t_1 \in q_j & \dots & w_{ij}|t_2 \in q_i, t_n \in q_j \\ \dots & \dots & \dots \\ w_{ij}|t_n \in q_i, t_1 \in q_j & \dots & w_{ij}|t_n \in q_i, t_n \in q_j \end{bmatrix} \quad (3)$$

Here, $MTF_{i,j}$ is the probability of transitioning from quantile bin q_i to q_j . Specifically, $MTF_{i,j}|i-j|=k$ denotes the probability of transitioning between points with a time interval of k .

c: RECURRENCE PLOTS (RP)

A Recurrence Plot (RP) is a binary depiction of a time series, revealing temporal correlations. Given a time series $T = [t_1, t_2, \dots, t_N]$, state vectors $\vec{t}(i) = [t_i, t_{i+d}, \dots, t_{i+(m-1)d}]$ are calculated using Time Delay Embeddings, for $i \in 1, 2, \dots, N - (m-1)d$, where d is temporal delay and m is trajectory dimension. The set $\{\vec{t}(i)\}$ represents the system's trajectory in the reconstructed phase space, denoting the evolution of the system's states during time. $RP_{i,j}$ becomes 1 if Euclidean distance between $\vec{t}(i)$ and $\vec{t}(j)$ is $\leq \epsilon$. For $K = N - (m-1)d$, the Recurrence Plot matrix [25], $K \times K$,

is computed according to Equation 4:

$$R_{i,j} = \begin{cases} 1, & \text{if } \|\vec{t}(i) - \vec{t}(j)\| \leq \epsilon \\ 0, & \text{else,} \end{cases} \quad \forall i, j \in 1, 2, \dots, K \quad (4)$$

d: DEEP LEARNING METHODS AS IMAGE FEATURE EXTRACTORS

Images hold diverse information like patterns, colors, textures, and shapes, including the ones generated by GAF, MTF, and RP methods. Leveraging pre-trained neural networks via transfer learning has exhibited substantial potential in diverse domains due to robust generalization and feature learning [53]. In this work, we utilized the CNN ResNet-152 and a large Vision Transformer (ViT) networks, both trained on ImageNet [54] via transfer learning, for feature extraction of GAF, MTF, and RP generated images.

ResNet [55] is a Convolutional Neural Network series designed for image recognition. ResNet-152, with up to 152 layers, overcomes gradient vanishing by using skip connections. These connections enhance gradient flow during backpropagation, enabling effective training of deep networks. The output for feature extraction is from the last fully connected layer.

Vision Transformers (ViT) [56] excel in object detection and image classification tasks. They employ self-attention to focus on key input parts concurrently. The input image transforms into patches, processed by an encoder generating hidden states. A decoder produces output tokens, with layers of self-attention and feedforward networks. Masked self-attention in the decoder attends to prior tokens. For feature extraction, we utilized the *CLS* token [57].

C. RE-RANKING METHODS BASED ON UNSUPERVISED DISTANCE LEARNING

The feature vectors associated with dataset samples can vary in spatial distribution within the feature space, potentially leading to severe impacts on machine learning tasks. On the other hand, unsupervised distance learning algorithms can improve retrieval and machine learning effectiveness [31], [34] by enhancing sample representations based on their rearrangement in the feature or distance space. These algorithms use a function f on ranked lists $\mathcal{R} = \{\tau_1, \tau_2, \dots, \tau_n\}$, calculated based on pairwise distances, to redefine distance/similarity measures and generate more global measurements, taking into account the dataset manifold. Based on the new distances, more effective ranked lists R_e are obtained. We employ four recent unsupervised methods, selected based on previous positive outcomes in image retrieval and classification tasks [34], [58], [59], [60]. Table 2 lists the re-ranking methods included in the comparative study, with their underlying principles and asymptotic complexity. Their descriptions are provided next.

TABLE 2. Summary of the utilized re-ranking methods, with the main data structures utilized in the algorithms.

Re-Ranking Methods	Acronym	Principle	Asymptotical Complexity
Log-based Hypergraph of Ranking References	LHRR	Hypergraphs	$O(n)$
Rank Diffusion Process with Assured Convergence	RDPAC	Diffusion	$O(n)$
Breadth-First Search Tree of Ranking References	BFS	Graphs	$O(n)$
Rank Flow Embedding	RFE	Embeddings	$O(n)$

1) LOG-BASED HYPERGRAPH OF RANKING REFERENCES (LHRR)

Hypergraphs generalize graphs with hyperedges, which connect multiple vertices and are capable of representing higher-order similarity relationships. The LHRR method uses a hypergraph model from ranking data, where hyperedges provide contextual dataset representation, enhancing similarity measurement efficiency through the product of hyperedge similarities [31]. The method is composed of five main steps:

- 1) A reciprocal rank normalization is performed, improving the symmetry of the k -neighborhood relationships with the similarity function $\rho_n(i, j) = 2L - (\tau_i(j) + \tau_j(i))$;
- 2) An hypergraph $G = (V, E, w)$ is constructed, where V is the set of vertices, E is the hyperedges set and w , the weights. Let e_i be and hyperedge defined for each object o_i , and v_j a vertex. The association between both is given by Equation 5, where $w_p(i, x) = 1 - \log_k \tau_i(x)$ is a weight function of relevance to o_x based on its position in τ_i . The weight of an hyperedge is given by $w(e_i) = \sum_{j \in \mathcal{N}_h(i, k)} h(i, j)$, where \mathcal{N}_h is the hypergraph neighborhood set;

$$r(e_i, v_j) = \sum_{o_x \in \mathcal{N}(i, k) \wedge o_j \in \mathcal{N}(x, k)} w_p(i, x) \times w_p(x, j) \quad (5)$$

- 3) Obtaining pairwise similarity information is necessary for ranking tasks. A pairwise similarity matrix \mathbf{S} is computed as follows: The similarity per common hyperedges is obtained by $S_h = HH^T$, where H is the incidence matrix. The similarity per vertices in common hyperedges is obtained with $S_v = H^T H$. Finally, S is given by the Hadamard product: $S = S_h \circ S_v$;
- 4) The Cartesian product of hyperedge elements is performed to maximize similarity information. A pairwise similarity function is given by $p(e_q, v_i, v_j) = w(e_q) \times h(e_q, v_i) \times h(e_q, v_j)$. Let e_q^2 be a cartesian product of elements belonging to the same hyperedge e_q , the similarity measurement based on the cartesian product is defined by the matrix \mathbf{C} , where:

$$c(i, j) = \sum_{e_q \in E \wedge (v_i, v_j) \in e_q^2} p(v_i, v_j) \quad (6)$$

- 5) Hypergraph and cartesian product informations are combined in $\mathbf{W} = \mathbf{C} \circ \mathbf{S}$, from where a new set of ranked lists is obtained. This process can be iteratively repeated.

Considering the described operations are performed in top- L ranked lists and sparse matrices, the overall time complexity of LHRR is $O(n)$.

2) RANK DIFFUSION PROCESS WITH ASSURED CONVERGENCE (RDPAC)

RDPAC [58] is an unsupervised distance learning algorithm with low computational complexity that approximates a diffusion process by exploring ranking information and assuring convergence. The method comprises four main steps:

- 1) Distance calculation based on ranking, where weights are attributed to positions, based on Ranked-Biased Overlap (RBO) [61]. Let s be the size of the ranked list and \mathbf{W}_s the computed affinity matrix. Each position is calculated based on $w_{sij} = p^{\tau_i(j)}$, if $\tau_i(j) \leq s$, otherwise 0. Given k the number of nearest neighbors, n the dataset size, $k < L \ll n$ and $s = L$, during rank diffusion, we assume $s = k$, and during normalization, we have $s = L$, obtaining sparse matrices \mathbf{W}_L and \mathbf{W}_k ;
- 2) A pre-diffusion rank normalization is performed, fixing the typical asymmetry of ranked lists ($w_{sij} \neq w_{sji}$). A symmetric version of the affinity matrix is given by $\mathbf{W}_L = \mathbf{W}_L + \mathbf{W}_L^T$. Ranking information is updated based on the new matrix, and the new ranked list is employed in the diffusion process. \mathbf{W}_k is calculated based on the ranked list and it is further column-wise normalized: $w_{ij}^{(t)} = w_{ij}^{(t)} / (\epsilon + \sum_{c=1}^n w_{jc}^{(t)})$;
- 3) The similarity information diffusion process with assured convergence is conducted iteratively over transition matrix \mathbf{P} , where $\mathbf{P}^{(1)} = \mathbf{W}_k$ and $\mathbf{P}^{(t+1)} = \alpha \mathbf{P}^{(t)} \mathbf{W}_k^T + (1 - \alpha)I$, where $\alpha \in [0, 1]$ and I is the identity matrix. The transpose \mathbf{W}_k^T is used for considering the multiplication among corresponding rank similarity scores;
- 4) An additional post-diffusion reciprocal analysis is performed, where information about the rank similarity of two elements i and j is explored. Given θ , the number of iterations in the previous step, the post diffusion step is computed with $\mathbf{R} = \mathbf{P}^\theta \mathbf{W}_k$, refining rankings with reciprocal references.

Given that operations are performed on top- k elements and the matrices are sparse, the final asymptotic complexity of RDPAC is $O(n)$.

3) BREADTH-FIRST SEARCH (BFS) TREE OF RANKING REFERENCES

The BFS-Tree of Ranking References [60] employs a Breadth-First Search tree on a graph of ranking references. This method has a high capacity to represent similarity information between the query and its k neighbors in a dataset. Objects occurring at different levels of the structure possibly exhibit higher similarity, while images occurring infrequently may indicate non-relevant elements or noise. The algorithm is composed of four main steps:

- 1) A ranking normalization pre-processing step, where the symmetry of k -neighborhood is improved by exploring mutual and reciprocal neighborhoods, i.e., rank positions from both rank references and the most pessimistic estimation, obtained with the maximum position, respectively. The mutual rank distance is calculated with $\rho_m(i, j) = \tau_i(j) + \tau_j(i)$, where $\tau_i(j) \leq L$. After the first re-ranking, the reciprocal rank distance is calculated by $\rho_r(i, j) = \max(\tau_i(j), \tau_j(i))$;
- 2) The BFS-Tree of ranking references is constructed starting from a query object root. The first level includes top- k similar objects according to rankings. Subsequent levels stem from top- k objects in the first level's ranking references. Edges are calculated using Ranked-Biased Overlap (RBO) [61]: $w(q, i) = (1 - p) \sum_{d=1}^k p^{d-1} \times (|\mathcal{N}_k(q) \cap \mathcal{N}_k(i)|)/d$;
- 3) The similarity between any node and the query root is calculated based on the Equation 7:

$$s(x_y) = \begin{cases} 1, & x = q \\ w(q, x_q), & y = q \wedge obj_x \in \mathcal{N}_k(q) \\ w(q, y_q) \times w(y_q, x_y), & x, y \neq q \wedge obj_y \in \mathcal{N}_k(q) \\ & \wedge obj_x \in \mathcal{N}_k(y) \\ 0, & x_y \notin V_q \end{cases} \quad (7)$$

After the similarities between the root and other nodes are defined, the similarities between every two objects $obj_i, obj_j \in V_q$ is defined by $\sigma_q(i, j) = \sum_{x, y \in V_q} s(i_x) \times s(j_y)$, and a more global similarity is calculated by $\sigma_a(i, j) = \sum_{obj_q \in \mathcal{C}} \sigma_q(i, j)$, taking into account all trees where obj_i and obj_j occur.

- 4) A single rank diffusion iteration is performed to improve the effectiveness of the method, post-processing the similarity function σ_a . The new function $\sigma_r(i, j)$ (Formally in Equation 8) considers the similarity to the common top- L positions of obj_i and obj_j

$$\sigma_r(i, j) = \sum_{x \in \mathcal{N}_L(i) \cap \mathcal{N}_L(j)} \sigma_a(i, x) \times \sigma_a(j, x) \quad (8)$$

Rank normalization is performed for top- L elements, and BFS-Tree analysis is performed for top- k elements, resulting in constant asymptotic complexity for each query and $O(n)$ when performing BFS-Tree in the whole dataset.

4) RANK FLOW EMBEDDING (RFE)

RFE is an algorithm proposed for both retrieval and classification tasks in unsupervised and semi-supervised scenarios, considering contextual information to obtain better results through stepwise improvement of rank-based similarity information [34]. RFE comprises five steps:

- 1) The first step is a ranked list normalization considering a sigmoid score based on reciprocal ranked list positions. The new similarity is obtained with $\rho_n(i, j) = \sigma(i, j)^2 \times \sigma(j, i)$, with $\sigma(x, y) = 1 - 1/(1 + e^{-\alpha(\tau_x(y) - k/2)})$, where α is a constant. Ranked lists are updated based on new distances;
- 2) Then, a re-ranking step is performed utilizing a hypergraph structure, built as described in step 2 of Section III-C1. Similarity is calculated by $\rho_h(i, j) = a_{ij}/\tau_i(j)$, with $A = HH^T$, updating the ranked list and repeating the procedure iteratively. The obtained hypergraph is also utilized for computing embeddings;
- 3) A re-ranking step based on the Cartesian product is performed. Steps are similar to step 4, described in Section III-C1;
- 4) A graph of Connect Component is constructed based on the h-embeddings computed after the Cartesian product operation, grouping objects with high global similarity, which are adjusted based on elements belonging to the same Connect Component;
- 5) More effective embeddings are computed based on the element similarity and its identified Connected Components, for semi-supervised classification tasks. This step is not necessary for time series retrieval.

Similarly to the previous algorithms, the operations are performed in the top- L elements and sparse matrices are utilized, resulting in $O(n)$ time complexity.

IV. EXPERIMENTAL EVALUATION

This section addresses the experimental protocol designed to provide a comparative study of the proposed approach for time series retrieval, considering representation and re-ranking methods. Section IV-A presents the time series datasets employed. Section IV-B, in turn, describes the effectiveness measures used in the assessment of retrieval results. Section IV-C covers the adopted parameter settings and the implementation details, while Section IV-D discusses the baselines.

A. DATASETS

Table 3 summarizes the datasets employed in the comparative study. We utilized six univariate time series datasets from the UCR Archive [62], which were z-normalized. These datasets were chosen based on factors like class number, domain variation, and size variation to ensure the versatility, robustness, and generality of our model. The datasets are: **Cylinder-Bell-Funnel (CBF)**: A simulated dataset with 930 elements of size 128, divided into three classes. Each class includes standard normal noise with a varying offset [63]. **GunPoint**: This dataset contains 200 time series,

each with a length of 150, capturing the movement of drawing a gun or pointing a finger [64]. **Beef**: Comprising 60 samples of silverside beef, divided into five classes. One class refers to pure beef, and the others are adulterated with different ingredients. Each series has a length of 470 [65]. **Rock**: Contains 70 spectral reflectance series from different types of rocks, each with a length of 2844, aiming to distinguish rock types [66]. **Electric Devices**: With 16637 time series, each having a length of 96, categorized into seven classes. These series represent daily power consumption data from various household devices [67]. **Yoga**: This dataset includes 3300 time series, each with a size of 426, generated from yoga pose transition videos. The goal is to distinguish between male and female actors based on the time series data [30].

B. EFFECTIVENESS MEASURES

This section presents the effectiveness measures used in the comparative study.

TABLE 3. Summary of the datasets, considering the sizes, mean number of relevant elements per query, and the standard deviation of the relevant elements per query.

Dataset	Queries	Mean	STD
Beef	60	12.00	0.00
CBF	930	310.00	0.00
GunPoint	200	100.00	0.00
Electric Devices (EDev)	16637	2376.71	1251.75
Rock	70	17.50	5.68
Yoga	3300	1650.00	120.00

1) PRECISION

Precision aims to evaluate the proportion of relevant items among all the retrieved items. It is defined as follows in Equation 9:

$$P = \frac{TP}{TP + FP}, \quad (9)$$

where TP represents true positives (correctly retrieved items) and FP represents false positives (incorrectly retrieved items).

2) RECALL

Recall measures the proportion of relevant instances that were retrieved among all the relevant instances. Recall is defined as follows in Equation 10:

$$R = \frac{TP}{TP + FN}, \quad (10)$$

where TP represents true positives and FN represents false negatives (correct instances that were not retrieved).

3) MEAN AVERAGE PRECISION (MAP)

Mean Average Precision (mAP) is a common metric for evaluating the effectiveness of ranked lists obtained in information retrieval tasks. This metric calculates the average of the Average Precision (AP) for each query [68]. The mAP

is defined in Equation 11:

$$mAP = \frac{\sum_{l=1}^Q AP(q_l)}{Q}, \quad (11)$$

where Q is the number of queries and q_l represents a query l . The Average Precision (AP), as defined in Equation 12, is the average precision for a recall varying from 0 to 100% [68]:

$$AP = \frac{1}{N_r} \sum_{i=1}^d \left(\frac{r_i}{i} \sum_{j=1}^i r_j \right), \quad (12)$$

where N_r is the number of relevant items in a collection for a query q and r_i represents the relevance (irrelevant = 0 or relevant = 1) of the i -th item in a ranked list of size d .

C. IMPLEMENTATION DETAILS AND PARAMETER DEFINITION

In this section, we outline the parameters, implementation details, and experimental protocol for the time series ranking model. The experiment comprises three stages: feature extraction, data ranking, and re-ranking. These stages were implemented using the Python programming language.

- 1) **Feature Extraction:** All feature extractors from Section III-B were employed. For time series imaging methods, the pyts [69] implementation with default parameters was used, and images were created and saved using the matplotlib library [70]. The colormap parameters were set to 'rainbow' for GAF and MTF methods and 'binary' for the RP method. The neural network architectures were applied to the generated images based on previous research results [19], [71]. Angles were measured in degrees in the Beam Angle Statistics algorithm, and the parameter k_b is set heuristically, with $k_b = z \times n \times \gamma$. In the formula, n is the length of the time series, z is the mean standard deviation of the dataset, calculated from individual time series standard deviations s_i , and the scaling factor $\gamma = 0.09$ was determined empirically during the experiments. The proposed approach satisfied the choice of k_b for five of the six datasets. It was unsuitable for Rock, which contains a time series of length 2844, significantly longer than those in the other datasets, resulting in $k_b = 3904$, producing an empty set of features due to exceeding the available dimensionality. $k_b = 30$ was manually set in this scenario. The application of the BAS algorithm was also demonstrated in [5]. For the DFT model, the pyts implementation [69] was utilized with parameters: $n_coefs = 0.4$, $norm_mean = True$, $norm_std = True$. PyWavelet [72] was used for the DWT model with parameters: $wavelet = 'haar'$ and $mode = 'sym'$, and results were separated into Approximation and Detail. The application of DFT and DWT methods was inspired by the study by Chan et al. [48]. The ROCKET algorithm, a recent method for time series classifica-

tion, was employed using the pyts implementation [69] with default parameters.

- 2) **Ranking:** After obtaining feature vectors for the time series, we calculate distance matrices using the Euclidean Distance. Each matrix row refers to a dataset element, sorted to form ranked lists, resulting in the set \mathcal{R} as defined in Section II. For each dataset, all the time series were used as queries.
- 3) **Re-ranking with Unsupervised Distance Learning Algorithms:** The unsupervised distance learning algorithms (discussed in Section III-C) were applied for post-processing ranking results. We aim to enhance distance measures by considering the dataset structure and improving retrieval outcomes. In this scenario, metrics other than Euclidean could be used, such as Minkowski, Chebyshev, Manhattan, Cosine, among others, as the main goal is the improvement of ranking results by enhancing the distance measurements. We used the Unsupervised Distance Learning Framework (UDLF) [73] implementation for these methods. We evaluated retrieval outcomes using Mean Average Precision (mAP), Precision@ x , and Recall@ x metrics both before and after re-ranking. Our parameter settings mostly followed defaults, with $k = 15$ for neighborhood analysis. For the ranked list size, denoted by L , we used $L = 1,000$ or the size of the dataset ($L = n$) when $n < 1,000$. To ensure consistent ranked list lengths across re-ranking methods, we set $L_{MULT} = 1$ for the RDPAC method. The effectiveness measures considered the value of x ranging from 1 to L in P@ x and R@ x metrics.

D. BASELINES

To assess the impact of time series representation approaches and re-ranking methods, we established comparative results. First, a ranked list is computed based on the time series raw values and the Euclidean distance, without employing any representation approach or re-ranking method. This ranking is referred to as the *initial* ranking. Subsequently, we construct *baselines* for each dataset, starting with the initial ranking and then applying different re-ranking methods.

V. RESULTS

This section presents and discusses the results of the proposed methodology. Section V-A presents a quantitative analysis of the mAP results, while Section V-B discusses qualitative results through the analysis of produced rankings. Section V-C provides an analysis of cases of success and failure, and, finally, Section V-D assesses scalability and efficiency analysis. In VI, complementary results regarding the assessment of methods in terms of the precision and recall metrics are presented.

A. RETRIEVAL RESULTS—QUANTITATIVE ANALYSIS

Table 4 presents the mAP results, considering different combinations of time series representation and re-ranking

methods on all datasets. Each column refers to a feature descriptor, and the comparisons are made by line. The best results for each feature extractor are highlighted in bold. For each dataset, a line relates to a re-ranking method, and the best results for each dataset are highlighted in red. We performed the Wilcoxon statistical test [74] comparing the best results, highlighted in red, with the others for the same dataset. The symbol ‘*’ is used for results for which no significant statistical differences were observed. The results related to the *initial* ranking are highlighted in italics and blue, for comparison. There are reported mean values (last column and last line) for the descriptors and re-ranking approaches.

For the Beef dataset, the most favorable outcome was observed for the combination of DWT-Detail and RFE, highlighted in red (Highest gains in terms of the re-ranking method, 11.27%, were observed for the combination of Moment and RFE, when compared with the Moment results with no re-ranking method). Conversely, searches on the CBF dataset predominantly benefited from the combination of the RDPAC re-ranking method with the Moment feature extractor and achieved the best results when considering the combination of DWT-Approximation and RDPAC (28.80% gain over the no re-rankings results). In the case of the GunPoint dataset, image-based methods proved to be superior. RP-ResNet coupled with RDPAC led to the best results (GAF-ViT in conjunction with BFSTREE yielded the most considerable improvement, 11.30% gain). In the Rock dataset, the combination of DWT-Detail and LHRR consistently delivered the best results, with the highest observed gains over all datasets (31.78%). For the Electric Devices dataset, the most effective combination was GAF-ResNet paired with RDPAC (The combination of the baseline and RFE delivered a gain of 23.46%. This is particularly interesting as it is the only noteworthy result involving the use of the baseline set of features). In the Yoga dataset, the gains were comparatively modest, at 3.58%, achieved with the ROCKET and RDPAC combination. Here, BAS and RFE yielded the best results.

Significant gains were observed in most scenarios of re-ranking, and losses occurred when the ranking efficacy was already low. This endorses the consistency of the pipeline, where better ranking leads to better re-ranking, allowing the use of other feature extractors, distance measurements, and unsupervised distance learning approaches. It is noteworthy that the RDPAC method consistently produced reliable results across the experiments on different datasets, while RFE yielded the best results for two different datasets (Beef and Yoga). While there was no consensus regarding the ideal feature extractor, some trends emerged. RP in conjunction with ResNet-152 yielded the highest average results, and DWT-Detail excelled in two datasets (Beef and Rock). The highest gains were observed in the Rock dataset. Noticing the general results, the Recurrence Plot approaches associated with RDPAC usually presented consistent results.

On the other hand, limitations on this approach can be observed. Concerns on efficiency and scalability can be

TABLE 4. mAP values considering different combinations of time series representations (column) and re-ranking methods (line) on all datasets. Results highlighted in bold refer to the best results for each time series representation across the different re-ranking methods. The result highlighted in red refers to the overall best result for the dataset. The symbol "*" is used for results for which no significant statistical differences were observed. "No re-rank" results refer to ranking with the Euclidean Distance, while "Baseline" results refer to experiments executed with re-ranking methods applied to the initial results. Both are highlighted in different tones of gray. Initial results are highlighted in blue and italics.

Specification		Descriptor													Mean
Dataset	UDL	Baseline	BAS	DFT	DWT Approx.	DWT Detail	ROCKET	GAF ResNet	MTF ResNet	RP ResNet	GAF ViT	MTF ViT	RP ViT	Moment	
Beef	No re-rank	<i>47.08*</i>	46.16	46.93*	47.03*	47.26*	34.79	40.83	37.53	40.86	42.75	40.02	45.60*	44.35	<i>43.17</i>
	LHRR	46.48	44.49	46.50	46.38	48.25	33.86	38.83	36.69	39.94	42.08	43.44	49.40	46.36	<i>43.29</i>
	RDPAC	47.49	44.26	47.64	47.57	49.58*	33.23	39.16	36.50	39.16	42.68	40.51	50.18*	47.92	<i>43.53</i>
	BFSTREE	46.83	44.50	46.78	46.86	49.05	33.78	41.79	36.56	39.93	42.81	43.92	48.69	46.44	43.69
	RFE	45.33	44.80	46.12	45.38	50.99	32.77	39.82	36.97	40.03	44.16	44.52	50.07*	49.35*	<i>43.65</i>
CBF	No re-rank	<i>64.47</i>	56.72	66.04	65.93	42.43	87.19	60.29	55.44	70.50	49.59	46.75	61.71	78.07	<i>61.93</i>
	LHRR	70.19	66.13	70.88	71.11	41.94	89.92	67.15	61.56	77.84	56.53	54.49	68.12	89.63	<i>68.12</i>
	RDPAC	81.45	68.23	84.04	84.92	38.09	95.43	73.48	66.69	84.22	62.29	58.34	72.49	98.02	74.44
	BFSTREE	73.30	66.28	75.36	76.33	46.59	90.71	68.98	63.06	78.72	57.54	54.95	70.72	94.57	<i>70.55</i>
	RFE	68.92	64.10	69.63	69.71	43.60	87.94	66.17	60.23	75.94	54.16	53.91	66.06	87.67	<i>66.77</i>
GunPoint	No re-rank	<i>62.41</i>	63.98	62.44	62.43	64.44	60.85	76.89	67.89	78.99	71.96	62.79	71.77	68.42	<i>67.33</i>
	LHRR	63.86	65.98	63.86	63.88	67.29	61.71	77.25	68.26	81.93	77.35	63.37	75.18	71.58	<i>69.35</i>
	RDPAC	64.70	67.72	64.61	64.58	66.73	61.90	79.17	70.85	86.28	78.66	63.57	77.25	71.74	70.60
	BFSTREE	63.97	66.73	64.14	64.05	69.84	62.22	79.70	69.33	80.84	80.09	64.12	77.70	71.16	<i>70.30</i>
	RFE	64.46	65.73	64.73	64.73	67.87	61.97	75.40	67.66	82.16	75.68	61.36	76.06	70.91	<i>69.13</i>
Rock	No re-rank	<i>58.71</i>	69.09	61.31	58.71	62.93	58.06	59.24	55.00	62.77	58.12	54.36	65.83	51.78	<i>59.68</i>
	LHRR	54.90	81.11*	66.76	54.90	82.93	53.34	64.40	61.37	62.94	62.94	59.23	65.99	53.52	<i>63.41</i>
	RDPAC	56.26	71.15	64.72	56.26	79.01*	53.05	63.21	60.46	63.33	64.54	61.53	65.03	52.98	<i>62.42</i>
	BFSTREE	57.06	80.27*	65.41	57.06	79.59	54.46	65.92	61.82	63.68	62.35	58.98	65.27	53.52	63.49
	RFE	56.29	78.03	65.81	56.29	67.84	54.11	65.57	58.09	62.39	63.55	54.90	64.65	52.49	<i>61.54</i>
EDev	No re-rank	<i>20.08</i>	24.66	24.94	23.54	14.80	37.56	41.17	34.97	40.24	40.05	37.01	35.97	34.99	<i>31.54</i>
	LHRR	24.75	21.30	23.95	23.24	14.52	35.24	39.13	34.30	39.00	38.16	34.50	36.69	33.22	<i>30.62</i>
	RDPAC	21.06	25.50	26.01	25.86	15.12	37.78	41.96	35.86	40.86	41.00	37.48	37.01	35.25	<i>32.36</i>
	BFSTREE	21.12	25.20	25.64	25.79	15.10	37.52	41.56	35.48	40.55	40.61	37.13	36.72	35.10	<i>32.12</i>
	RFE	24.79	26.05	28.05	27.27	17.17	37.60	41.63	35.82	40.89	40.77	36.81	37.51	35.21	33.04
Yoga	No re-rank	<i>30.30</i>	31.58	30.30	30.30	30.23	29.89*	30.50	28.77	30.60	30.17	27.93	29.83	30.71	<i>30.08</i>
	LHRR	30.41	31.24	30.41	30.41	29.82	30.18	30.81	28.86	30.52	30.77	28.57	30.00	30.81	<i>30.22</i>
	RDPAC	30.76	31.79*	30.75	30.76	30.32	30.96	31.45	29.50	31.44	31.03*	28.79	30.54	31.32	30.72
	BFSTREE	30.59*	31.71	30.58*	30.58*	30.38	30.56	31.33*	29.34	31.32*	30.95*	28.63	30.38	30.88	<i>30.56</i>
	RFE	30.47	31.86	30.46	30.47	30.41	30.19	31.18*	29.19	31.15*	30.73	28.62	30.35	30.93	<i>30.46</i>
Mean		48.62	51.21	50.83	49.41	46.47	51.29	53.46	48.47	55.63	51.47	47.02	54.09	54.30	

raised, and this issue, with possible overcomes, is discussed in Section V-D.

Our pipeline, while it does not make any assumption about stationarity of the time series, may be highly demanding on memory usage, by storing datasets, features, and rankings, especially in large-scale scenarios or real-time systems, where the latency is a crucial aspect [75]. Additionally, the parameter sensitivity during both the feature extraction and re-ranking steps, paired with the challenge of selecting the best combination between the two feature extractors and manifold learning approaches, are also challenging points in our work.

B. RETRIEVAL RESULTS—QUALITATIVE ANALYSIS

Figures 2, 3, 4, and 5 show examples of qualitative results of a query for the Rock dataset. The following images represent the classes of the time series encoded in different colors. The queries are highlighted with green boxes, while red boxes are used for the non-relevant results.

It is possible to observe that some representations led to improved results while others (time series raw values) did not. Also, it is possible to observe that the performance of the re-ranking algorithms was quite dependent on the representation used and ranking results, which can encode more or less useful information for ranking. For example, for the cases in

which the ranking produced by the representation contains enough relevant time series at the top-ranked positions (Figures 2 and 3), the use of the re-ranking methods led to very positive gains, with more relevant time series ranked at the top positions, as the ranking produced by DWT-Detail and BAS already contained relevant time series. On the other hand, Figures 4 and 5 show examples in which the ranking produced by the representation does not contain many relevant time series at the top-ranked positions (raw time series values and RP-ViT), and the use of the re-ranking methods did not lead to significant improvements.

C. RETRIEVAL RESULTS—ANALYSIS OF CASES OF SUCCESS AND FAILURE

In Table 4, it is possible to observe that certain methods exhibit better outcomes than others depending on the dataset. This section sheds light on some successful and unsuccessful cases identified.

A comparison between methods on the Beef and GunPoint datasets was performed, as they are similar datasets, composed of a small number of time series. The use of re-ranking methods on the GunPoint dataset consistently delivered positive and noteworthy improvements. On the other hand, the results observed on the Beef dataset were not always positive for the use of re-ranking methods. The analysis is

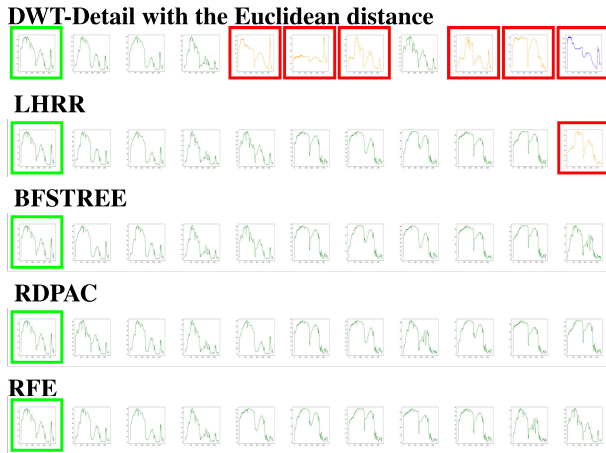


FIGURE 2. Visual ranking results on Rock [66] dataset for DWT-Detail representations. The query is highlighted with green boxes, while red boxes are used for the non-relevant results.

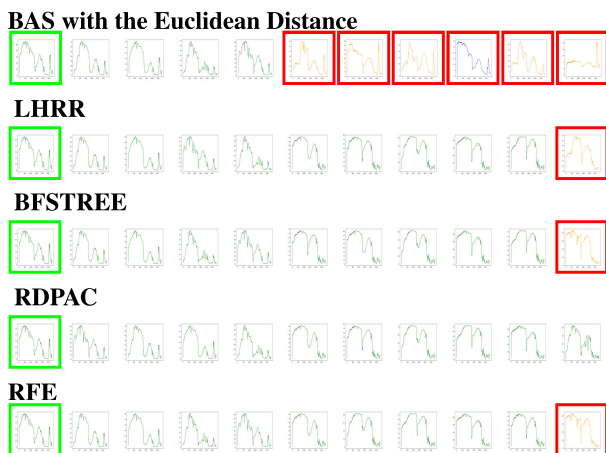


FIGURE 3. Visual ranking results on Rock [66] dataset for BAS representations.



FIGURE 4. Visual ranking results on Rock [66] dataset when the time series raw values are used.

conducted considering the Precision@10, Recall@10, and Average Precision metrics of each query of the dataset.

In Figure 6, we analyze the performance of the ROCKET feature extractor on the Beef dataset. In Table 4, it is



FIGURE 5. Visual ranking results on Rock [66] dataset for RP-ViT representations.

possible to observe that there were no gains on mAP after re-ranking. When using this representation method, we observed a decline in mAP across all re-ranking methods. It is worth noting that all metrics maintained an average of around 30%. The initial 10 queries, observed in the x-axis, and queries between 30 and 40 display a better, above-average performance, especially when considering the LHRR and RFE methods. Drops are observed throughout the remaining queries, with a noticeable inferior performance of RDPAC. The below-average performance in most queries, associated with low metric values, may explain why the re-ranking methods did not perform as expected.

In Figure 7, we present the metrics obtained from ranking the GunPoint dataset using the RP-ResNet152 combination for feature extraction. Observing Table 4, we see that this combination resulted in the best mAP value when considering the re-ranking with RDPAC. With this representation method, we not only observe overall gains in all re-ranking scenarios, in which most queries presented P@10, R@10, and mAP with above-average values, but also note that the highest mAP value was achieved when using RDPAC. The average results for mAP and precision metrics were consistently high, and there were various peaks and drops for the mAP metric. Precision and recall metrics exhibit stability, with very few declines. Consistently, the re-ranking methods delivered superior results in comparison with the Euclidean Distance ranking. The observed stability may be an important factor for the gains after re-ranking.

In Figure 8, we present a comparison of results obtained using the GAF-ViT feature extraction methods, which led to positive mAP gains in both datasets. Interestingly, we observe similar patterns to those seen in Figures 6 and 7. Notably, for the Beef dataset, the average results were higher compared to the previous example, and all unsupervised distance learning methods performed better than the original, especially RFE. For the GunPoint dataset, BFSTREE yielded better results, while RFE and RDPAC produced some losses for a few

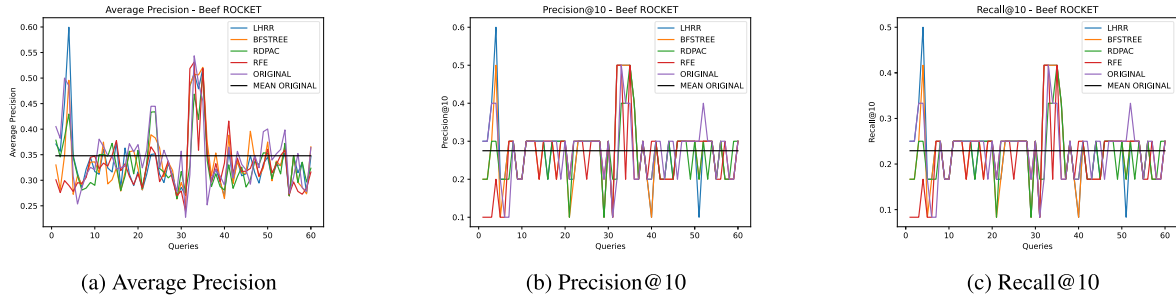


FIGURE 6. Measurements for the Beef dataset, considering the ROCKET feature extractor.

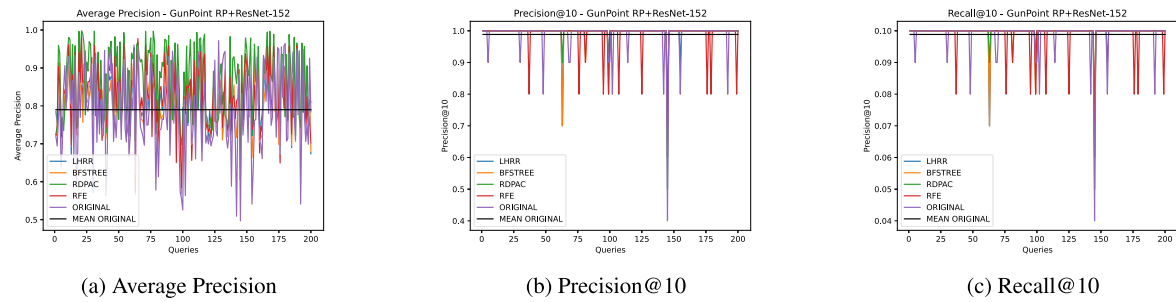


FIGURE 7. Measurements for GunPoint dataset, considering the RP+ResNet feature extractor.

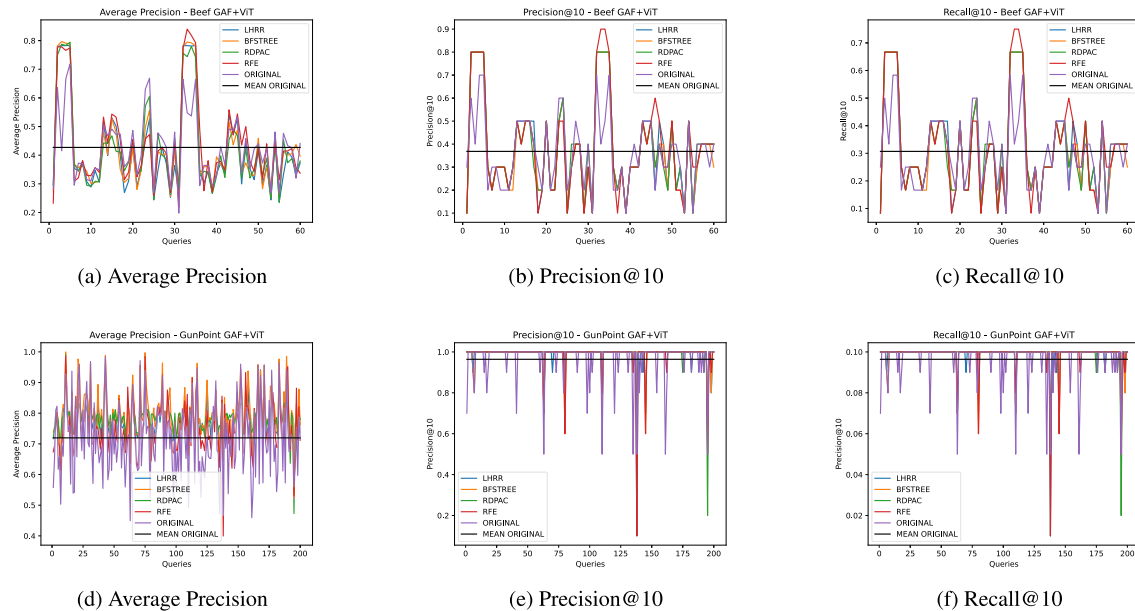


FIGURE 8. Comparison between datasets for GAF+ViT feature extractor.

queries, despite the consistently average positive results of all re-ranking methods.

The obtained results suggest that positive outcomes are achieved through effective re-ranking when there is sufficient contextual information available in the input ranked lists computed based on the Euclidean distance, i.e., better re-ranking results are achieved through better ranking results. Otherwise, it is a challenging task to obtain relevant gains when the original ranking presents low values of precision,

recall, and average precision, i.e., when the input ranked lists do not contain many relevant time series at the top positions. In those cases, there might not be enough information available for re-ranking methods to enhance the outcomes.

D. RETRIEVAL RESULTS-EFFICIENCY ANALYSIS AND SCALABILITY

Time experiments were conducted to evaluate the efficiency of retrieval tasks and the scalability. The implementa-

TABLE 5. Time measurements of executions, in seconds, of the three biggest datasets. Mean and standard deviation of 5 executions, querying all elements of the dataset.

Specification		Descriptor													
Dataset	UDL	Baseline	BAS	DFT	DWT Approx.	DWT Detail	ROCKET	GAF ResNet	MTF ResNet	RP ResNet	GAF ViT	MTF ViT	RP ViT	Moment	Mean
CBF	LHRR	0.63 ± 0.01	0.73 ± 0.01	0.62 ± 0.00	0.63 ± 0.01	0.96 ± 0.01	0.66 ± 0.01	0.74 ± 0.01	0.74 ± 0.0	0.73 ± 0.01	0.78 ± 0.01	0.78 ± 0.01	0.75 ± 0.0	0.7 ± 0.01	0.73
	RDPAC	5.41 ± 0.06	2.73 ± 0.04	5.46 ± 0.05	5.58 ± 0.05	2.78 ± 0.05	5.42 ± 0.06	2.74 ± 0.02	2.74 ± 0.06	2.77 ± 0.06	2.73 ± 0.05	2.71 ± 0.03	2.73 ± 0.08	2.76 ± 0.07	3.58
	BFSTREE	0.95 ± 0.0	0.97 ± 0.01	0.95 ± 0.0	0.95 ± 0.01	1.01 ± 0.01	0.96 ± 0.01	0.98 ± 0.0	0.98 ± 0.0	0.97 ± 0.01	0.97 ± 0.01	0.97 ± 0.01	0.98 ± 0.01	0.97 ± 0.01	0.97
	RFE	3.52 ± 0.05	6.16 ± 0.01	3.23 ± 0.05	3.29 ± 0.03	11.35 ± 0.03	4.52 ± 0.02	6.76 ± 0.02	7.1 ± 0.04	6.54 ± 0.01	7.7 ± 0.02	7.72 ± 0.06	7.15 ± 0.04	6.08 ± 0.01	6.24
EDev	LHRR	97.34 ± 0.32	99.88 ± 0.2	96.38 ± 0.53	95.5 ± 0.6	97.69 ± 0.52	93.72 ± 0.48	93.92 ± 0.46	93.13 ± 0.38	93.2 ± 0.58	93.72 ± 0.36	92.71 ± 0.65	92.66 ± 0.5	92.96 ± 0.64	94.83
	RDPAC	404.2 ± 0.84	406.75 ± 0.61	334.49 ± 0.24	330.04 ± 0.38	358.84 ± 0.48	369.61 ± 0.31	362.58 ± 0.67	340.31 ± 0.63	366.21 ± 1.2	375.48 ± 0.85	341.52 ± 0.61	359.88 ± 1.11	379.23 ± 0.16	363.78
	BFSTREE	133.39 ± 0.56	131.99 ± 0.42	126.71 ± 0.51	126.24 ± 0.59	133.16 ± 0.36	125.35 ± 0.3	123.1 ± 0.44	123.73 ± 0.62	124.15 ± 0.4	123.51 ± 0.29	123.05 ± 0.15	122.22 ± 0.28	124.24 ± 0.4	126.22
	RFE	290.06 ± 0.85	352.66 ± 0.65	292.44 ± 0.59	265.84 ± 0.52	284.28 ± 0.54	276.59 ± 1.19	303.55 ± 0.79	292.03 ± 0.84	293.17 ± 1.04	327.84 ± 0.42	324.11 ± 5.41	297.06 ± 1.66	326.71 ± 0.81	302.03
Yoga	LHRR	5.03 ± 0.03	5.16 ± 0.07	5.04 ± 0.06	5.11 ± 0.07	5.11 ± 0.03	5.11 ± 0.03	5.47 ± 0.05	5.69 ± 0.1	5.38 ± 0.12	5.32 ± 0.01	5.71 ± 0.05	5.25 ± 0.06	4.91 ± 0.06	5.25
	RDPAC	40.19 ± 0.2	40.84 ± 0.28	40.16 ± 0.22	40.28 ± 0.19	38.92 ± 0.16	36.52 ± 0.04	40.62 ± 0.15	40.7 ± 0.19	39.68 ± 0.08	41.12 ± 0.12	40.86 ± 0.26	40.65 ± 0.14	39.9 ± 0.17	40.03
	BFSTREE	8.25 ± 0.08	8.29 ± 0.08	8.27 ± 0.04	8.2 ± 0.07	8.12 ± 0.06	8.24 ± 0.04	8.36 ± 0.05	8.42 ± 0.06	8.34 ± 0.06	8.39 ± 0.06	8.37 ± 0.04	8.26 ± 0.04	8.05 ± 0.05	8.27
	RFE	18.7 ± 0.15	20.17 ± 0.09	18.65 ± 0.09	18.61 ± 0.06	22.04 ± 0.1	19.27 ± 0.05	26.85 ± 0.06	32.11 ± 0.11	25.4 ± 0.08	24.44 ± 0.08	33.17 ± 0.16	23.86 ± 0.08	19.56 ± 0.04	23.30
Mean		13.48	14.23	13.46	13.41	16.70	13.76	17.61	20.27	16.87	16.42	20.77	16.06	13.81	

TABLE 6. P@5 values.

Specification		Descriptor													
Dataset	UDL	Baseline	BAS	DFT	DWT Approx.	DWT Detail	ROCKET	GAF ResNet	MTF ResNet	RP ResNet	GAF ViT	MTF ViT	RP ViT	Moment	Mean
Beef	No re-rank	56.67*	54.67*	57.00*	56.67*	60.00*	40.33	50.00	41.00	48.67	51.67	44.33	57.00*	54.33	51.72
	LHRR	53.00	51.33	53.00	52.67*	59.33	39.00	48.67	39.67	47.67	51.33	52.67	60.00*	53.00	50.88
	RDPAC	57.00*	52.33	57.33*	57.00*	59.67*	35.33	48.67	41.33	51.00	51.67	50.33	61.00	58.00*	52.36
	BFSTREE	55.33*	49.67	55.00*	54.67*	57.33*	36.00	49.00	39.67	47.33	50.67	51.33	58.00*	53.67	50.59
	RFE	52.67	50.67	52.33	53.33	59.00*	34.67	50.00	41.67	44.67	52.33	54.00	57.67*	55.33	50.64
CBF	No re-rank	98.39	89.16	99.89*	99.81*	48.69	99.85*	91.76	88.69	96.26	84.24	80.30	93.68	99.63	96.90
	LHRR	100.0	96.97	100.0	99.91*	65.12	99.83*	91.38	90.19	96.06	85.42	80.60	91.59	99.87*	99.13
	RDPAC	99.94*	96.58	100.0	99.94*	60.11	99.94*	92.52	90.17	96.37	85.29	80.52	93.08	99.89	98.91
	BFSTREE	100.0	96.04	100.0	99.91*	65.48	99.91*	91.78	89.59	95.57	84.67	79.44	92.86	99.89*	98.98
	RFE	100.0	96.32	100.0	99.91*	64.26	99.91*	91.46	89.78	95.53	84.41	79.55	92.43	99.83*	98.83
GunPoint	No re-rank	93.1	90.7	93.2	93.2	91.0	94.5	98.7*	88.7	99.7*	98.0*	81.4	99.2*	97.80	93.78
	LHRR	93.4	90.0	94.1	93.8	91.5	93.3	99.5*	87.0	99.8	98.8*	82.7	99.2*	97.50	93.89
	RDPAC	92.4	89.1	92.7	92.8	92.6	94.7	99.2*	87.5	99.8	99.2*	81.2	99.7*	97.70	93.74
	BFSTREE	92.2	90.2	93.0	92.8	92.6	93.8	99.3*	86.5	99.4*	99.3*	81.3	99.4*	97.10	93.61
	RFE	92.7	90.3	92.7	93.1	91.7	94.5	97.9*	87.7	99.5*	98.6*	82.5	98.6*	97.60	93.65
Rock	No re-rank	78.00	81.43*	80.86*	78.00	80.57	74.00	75.43	66.00	76.57	79.71	69.43	79.43	69.14	76.04
	LHRR	74.29	86.29*	82.00*	74.29	87.43	65.71	80.00*	70.29	74.86	82.57*	66.57	78.29	67.71	76.35
	RDPAC	74.86	82.86*	82.86*	74.86	86.00*	69.43	77.71	69.43	73.43	79.71	73.14	76.00	68.86	76.15
	BFSTREE	75.71	86.00*	81.43*	75.71	84.57*	69.43	79.14*	70.00	75.14	82.29*	68.00	78.00	69.14	76.46
	RFE	75.43	86.86*	83.14*	75.43	83.43	69.43	81.43*	69.14	74.00	80.00*	66.29	77.71	65.71	76.51
EDev	No re-rank	74.48	70.64	72.30	73.14	61.91	81.36	84.94*	78.80	84.55	83.40	78.26	83.70	79.26	77.44
	LHRR	74.27	71.85	73.31	73.35	66.53	80.95	85.30	78.82	84.54	83.69	78.38	83.90	79.28	78.02
	RDPAC	74.07	71.09	72.41	72.67	64.73	80.74	85.04	78.86	84.41	83.60	78.03	83.80	79.04	77.57
	BFSTREE	74.12	71.71	73.45	73.73	65.65	80.86	85.30	78.98	84.51	83.51	78.43	83.86	79.27	77.95
	RFE	74.28	71.78	73.46	73.72	65.30	80.79	85.19*	78.63	84.43	83.47	78.37	83.83	79.26	77.88
Yoga	No re-rank	92.69	90.79	92.69	92.70	88.02	94.19	88.92	82.31	90.23	89.03	82.77	90.47	93.52*	89.87
	LHRR	92.42	90.65	92.47	92.41	91.14	93.98*	88.70	83.06	90.36	88.68	83.68	90.36	93.62	90.12
	RDPAC	92.48	90.76	92.45	92.38	90.59	94.05*	88.90	83.14	90.55	89.24	83.73	90.41	93.59	90.18
	BFSTREE	92.55	90.70	92.65	92.62	91.42	94.03*	88.44	83.35	90.48	88.73	83.30	90.68	93.76	90.21
	RFE	92.64	90.79	92.67	92.55	90.69	93.87*	88.60	82.80	90.39	88.87	83.16	90.19	90.42	89.82
Mean		81.64	80.61	82.61	81.57	75.21	79.28	82.10	74.76	82.19	81.40	73.79	83.80	82.09	

tion was evaluated on a machine with an Intel Core i5-12500H @ 2.5 GHz, 24 GB of RAM, and Ubuntu 24.04.1 LTS (64-bit). Table 5 presents the execution time, in seconds, of re-ranking results in the three biggest datasets (CBF, Yoga, and Electric Devices) we experimented with. Fastest results per dataset are highlighted in bold.

We can clearly notice that the impact of re-ranking approaches on time is bigger than the descriptor impact, and despite all methods having linear asymptotic complexity, LHRR stands with the fastest results, and RDPAC with the slowest. The feature extraction methods were less impactful on time experiments, but DTW Approximation stands with the fastest results, on average.

VI. CONCLUSION

In this study, we conducted an extensive examination of time series representations and re-ranking methods for time series retrieval tasks. The findings indicate that a proper combination of time series representation and re-ranking methods holds promise for time series retrieval, with favorable outcomes observed in the majority of scenarios. Nonetheless, it is worth noting that while results were mostly positive, the use of certain re-ranking methods led to minor losses (e.g., the Beef and Rock datasets, considering the ROCKET feature extractor). Upon closer investigation, these losses appear to be attributed to a limited amount of contextual information available for substantial re-ranking improvements. No single combination of time series retrieval

TABLE 7. P@10 values.

Specification		Descriptor													
Dataset	UDL	Baseline	BAS	DFT	DWT Approx.	DWT Detail	ROCKET	GAF ResNet	MTF ResNet	RP ResNet	GAF ViT	MTF ViT	RP ViT	Moment	Mean
Beef	No re-rank	43.17*	41.33	43.17*	43.17*	42.50	27.50	34.83	31.17	33.50	36.83	33.33	42.83*	40.17	37.96
	LHRR	42.17	40.17	42.00	42.17	47.33*	28.17	35.33	29.67	34.33	38.50	39.00	48.17*	44.33	39.34
	RDPAC	44.00*	40.33	44.33*	44.33*	48.50*	26.50	34.67	31.83	33.83	38.33	33.50	49.33	43.83	39.48
	BFSTREE	43.50*	38.83	43.33*	43.50*	47.83*	27.00	37.83	29.50	34.67	37.67	38.50	46.67*	43.50	39.41
	RFE	41.83	39.50	41.67	41.33	48.33*	25.67	34.50	28.17	33.17	38.83	39.33	47.17*	48.17	39.05
CBF	No re-rank	97.44	85.45	99.54*	99.48*	42.46	99.33	88.04	84.62	94.18	78.94	73.37	87.66	99.24	86.90
	LHRR	99.95*	95.66	99.95*	99.81*	58.83	99.59*	89.26	86.61	94.67	80.84	74.59	90.10	99.75	89.97
	RDPAC	99.86*	95.96	100.00	99.86*	51.91	99.74*	90.00	87.02	95.53	80.91	75.46	90.72	99.83	89.75
	BFSTREE	99.99*	95.40	100.00	99.82*	59.83	99.74*	89.04	86.20	94.17	80.06	73.96	90.00	99.74	89.84
	RFE	100.00	95.11	99.88*	99.73*	58.03	99.74*	88.49	85.69	94.05	79.70	72.67	88.76	99.72	89.35
GunPoint	No re-rank	87.20	85.50	87.30	87.45	86.85	88.85	97.40	83.80	98.85*	96.35	75.10	98.25*	96.25	89.94
	LHRR	88.80	84.85	88.80	89.05	89.55	90.05	98.85*	84.00	99.55*	98.45*	76.70	98.40*	96.40	91.03
	RDPAC	88.85	85.25	88.20	88.05	89.55	91.65	98.70*	83.95	99.65	99.30*	76.75	99.55*	96.75	91.24
	BFSTREE	88.85	85.70	89.25	89.05	91.50	90.75	98.85*	84.05	99.50*	98.85*	75.90	98.50*	96.10	91.30
	RFE	88.50	84.30	88.95	89.15	89.50	90.65	95.25	83.55	98.90*	98.45*	74.65	96.90	96.15	90.33
Roec	No re-rank	60.29	69.43	68.57	60.29	65.14	58.00	64.00	54.57	65.29	63.43	54.86	68.43	54.86	62.09
	LHRR	57.29	82.00*	77.43*	57.29	83.29	56.14	70.29	57.86	65.14	68.86	59.57	71.71	58.43	66.56
	RDPAC	56.86	75.14	74.57*	56.86	80.14	55.00	67.86	58.86	64.86	68.43	63.29	69.14	58.29	65.33
	BFSTREE	59.14	80.86*	75.29*	59.14	79.57*	55.57	70.86	60.43	65.14	68.43	59.86	70.43	58.14	66.37
	RFE	58.00	80.14*	74.86*	58.00	68.71	55.71	70.86	54.57	64.00	69.43	56.14	70.00	54.86	64.26
EDev	No re-rank	68.57	64.08	65.83	67.17	53.98	77.29	81.47	75.11	80.58	79.52	74.52	79.82	74.38	72.49
	LHRR	68.83	65.79	67.91	68.05	60.27	76.82	81.79*	74.89	81.02	80.08	74.22	80.28	74.61	73.43
	RDPAC	68.55	65.11	66.75	67.45	57.10	76.51	81.74*	74.87	80.87	80.03	73.99	80.25	74.36	72.89
	BFSTREE	69.03	65.86	67.89	68.64	58.05	76.91	81.87	74.87	81.09	79.96	74.23	80.33	74.71	73.34
	RFE	68.73	65.53	67.96	68.31	58.80	76.67	81.62*	74.46	80.67	79.66	74.14	80.03	74.23	73.14
Yoga	No re-rank	88.99	86.63	89.00	88.99	83.40	90.94*	84.38	76.73	85.73	84.57	77.20	85.91	89.98	85.58
	LHRR	89.28	86.92	89.28	89.26	87.68	91.17*	84.67	78.27	86.95	85.05	78.88	86.98	90.64	86.54
	RDPAC	89.21	87.02	89.26	89.16	86.96	91.25*	85.15	78.30	87.26	85.34	79.35	87.19	90.78	86.63
	BFSTREE	89.48	87.63	89.52	89.48	87.87	91.28	84.82	78.57	86.96	85.36	78.89	87.31	90.66	86.76
	RFE	89.49	87.35	89.56	89.49	87.11	90.92	84.59	77.59	86.61	84.88	78.35	86.64	93.30	86.61
Mean		74.53	74.76	77.00	74.45	68.35	73.50	76.23	68.33	76.69	74.84	66.34	78.58	77.07	

TABLE 8. R@5 values.

Specification		Descriptor													
Dataset	UDL	Baseline	BAS	DFT	DWT Approx.	DWT Detail	ROCKET	GAF ResNet	MTF ResNet	RP ResNet	GAF ViT	MTF ViT	RP ViT	Moment	Mean
Beef	No re-rank	23.61*	22.78*	23.75*	23.61*	25.00*	16.81	20.83	17.08	20.28	21.53	18.47	23.75*	22.64*	21.55
	LHRR	22.08	21.39	22.08	21.94	24.72*	16.25	20.28	16.53	19.86	21.39	21.94*	25.00*	22.08*	21.19
	RDPAC	23.75*	21.81	23.89*	23.75*	24.86*	14.72	20.28	17.22	21.25	21.53	20.97	25.42	24.17*	21.82
	BFSTREE	23.06*	20.69	22.92*	22.78*	23.89*	15.00	20.42	16.53	19.72	21.11	21.39	24.17*	22.36*	21.08
	RFE	21.94	21.11	21.81*	22.22*	24.58*	14.44	20.83	17.36	18.61	21.81*	22.50*	24.03*	23.06*	21.10
CBF	No re-rank	1.59	1.44	1.61	1.61	0.79	1.61	1.48	1.43	1.55	1.36	1.30	1.48	1.61	1.45
	LHRR	1.61	1.56	1.61	1.61	1.05	1.61	1.47	1.45	1.55	1.38	1.30	1.51	1.61	1.49
	RDPAC	1.61	1.56	1.61	1.61	0.97	1.61	1.49	1.45	1.55	1.38	1.30	1.50	1.61	1.48
	BFSTREE	1.61	1.55	1.61	1.61	1.06	1.61	1.48	1.45	1.54	1.37	1.28	1.50	1.61	1.48
	RFE	1.61	1.55	1.61	1.61	1.04	1.61	1.48	1.45	1.54	1.36	1.28	1.49	1.61	1.48
GunPoint	No re-rank	4.65	4.53	4.66	4.66	4.55	4.73	4.94*	4.43	4.99	4.90*	4.07	4.96*	4.89	4.69
	LHRR	4.67	4.50	4.70	4.69	4.57	4.66	4.98*	4.35	4.99	4.94*	4.13	4.96*	4.88	4.70
	RDPAC	4.62	4.45	4.63	4.64	4.63	4.73	4.96*	4.37	4.99	4.96*	4.06	4.99	4.89	4.69
	BFSTREE	4.61	4.51	4.65	4.64	4.63	4.69	4.97*	4.32	4.97*	4.97*	4.06	4.97*	4.86	4.68
	RFE	4.64	4.51	4.64	4.65	4.58	4.72	4.90*	4.38	4.98*	4.93*	4.12	4.93*	4.88	4.68
Roec	No re-rank	24.70	25.35	25.36	24.70	25.18	21.81	23.66	22.01	24.47	23.76	23.10	23.81	21.53	23.80
	LHRR	23.36	28.06*	26.01*	23.36	28.39	18.65	24.73*	23.55	23.33	24.93*	21.76	23.81	20.57	23.95
	RDPAC	23.97	27.50*	25.81*	23.97	28.11*	20.27	23.37	23.25	23.18	24.42	23.34	24.20	20.80	24.03
	BFSTREE	24.09	28.06*	25.25*	24.09	27.28*	20.24	23.70*	23.30	23.46	24.39*	22.84	24.20	21.91	24.00
	RFE	23.97	28.12*	25.61*	23.97	27.48*	20.46	25.14*	22.23	23.24	23.83	21.08	23.34	20.62	23.94
EDev	No re-rank	0.17	0.16	0.17	0.17	0.14	0.20	0.21	0.19	0.21	0.20	0.19	0.21	0.19	0.19
	LHRR	0.17	0.17	0.17	0.17	0.15	0.19	0.21	0.19	0.21	0.20	0.19	0.21	0.19	0.19
	RDPAC	0.17	0.17	0.17	0.17	0.15	0.19	0.21	0.19	0.21	0.20	0.19	0.21	0.19	0.19
	BFSTREE	0.17	0.17	0.17	0.17	0.15	0.19	0.21	0.19	0.21	0.20	0.19	0.21	0.19	0.19
	RFE	0.17	0.17	0.17	0.17	0.15	0.19	0.21	0.19	0.21	0.20	0.19	0.21	0.19	0.19
Yoga	No re-rank	0.28	0.28	0.28	0.28	0.27	0.29	0.27	0.25	0.27	0.27	0.25	0.27	0.28*	0.27
	LHRR	0.28	0.28	0.28	0.28	0.28	0.29	0.27	0.25	0.27	0.27	0.25	0.27	0.28*	0.27
	RDPAC	0.28	0.28	0.28	0.28	0.28	0.29	0.27	0.25	0.28	0.27	0.25	0.28	0.28*	0.27
	BFSTREE	0.28	0.28	0.28	0.28	0.28	0.29	0.27	0.25	0.28	0.27	0.25	0.28	0.29	0.27
	RFE	0.28	0.28	0.28	0.28	0.28	0.29	0.27	0.25	0.27	0.27	0.25	0.27	0.28*	0.27
Mean		8.93	9.24	9.20	8.93	9.65	7.09	8.59	7.68	8.41	8.75	8.22	9.22	8.50	

and re-ranking methods led to superior results for all datasets. In general, the best average results were associated with

RP + ResNet-152 feature extraction and RDPAC for the re-ranking approach. Still, these results suggest that their

TABLE 9. R@10 values.

Specification		Descriptor													
Dataset	UDL	Baseline	BAS	DFT	DWT Approx.	DWT Detail	ROCKET	GAF ResNet	MTF ResNet	RP ResNet	GAF ViT	MTF ViT	RP ViT	Moment	Mean
Beef	No re-rank	35.97*	34.44	35.97*	35.97*	35.42	22.92	29.03	25.97	27.92	30.69	27.78	35.69*	33.47	31.63
	LHRR	35.14	33.47	35.00	35.14	39.44*	23.47	29.44	24.72	28.61	32.08	32.50	40.14*	36.94	32.78
	RDPAC	36.67*	33.61	36.94*	36.94*	40.42*	22.08	28.89	26.53	28.19	31.94	27.92	41.11	36.53	32.90
	BFSTREE	36.25*	32.36	36.11*	36.25*	39.86*	22.50	31.53	24.58	28.89	31.39	32.08	38.89*	36.25	32.84
	RFE	34.86	32.92	34.72	34.44	40.28*	21.39	28.75	23.47	27.64	32.36	32.78	39.31*	40.14*	32.54
CBF	No re-rank	3.14	2.76	3.21*	3.21*	1.37	3.20	2.84	2.73	3.04	2.55	2.37	2.83	3.20	2.80
	LHRR	3.22*	3.09	3.22*	3.22*	1.90	3.21*	2.88	2.79	3.05	2.61	2.41	2.91	3.22	2.90
	RDPAC	3.22*	3.10*	3.23	3.22*	1.67	3.22*	2.90	2.81	3.08	2.61	2.43	2.93	3.22	2.90
	BFSTREE	3.23	3.08	3.23	3.22*	1.93	3.22*	2.87	2.78	3.04	2.58	2.39	2.90	3.22	2.90
	RFE	3.23	3.07	3.22*	3.22*	1.87	3.22*	2.85	2.76	3.03	2.57	2.34	2.86	3.22	2.88
GunPoint	No re-rank	8.72	8.55	8.73	8.74	8.68	8.88	9.74	8.38	9.89*	9.64	7.51	9.83*	9.63	8.99
	LHRR	8.88	8.48	8.88	8.90	8.95	9.00	9.89*	8.40	9.96*	9.85*	7.67	9.84*	9.64	9.10
	RDPAC	8.88	8.52	8.82	8.80	8.96	9.16	9.87*	8.39	9.97	9.93*	7.68	9.96*	9.68	9.13
	BFSTREE	8.89	8.57	8.92	8.90	9.15	9.07	9.89*	8.40	9.95*	9.89*	7.59	9.85*	9.61	9.13
	RFE	8.85	8.43	8.89	8.91	8.95	9.06	9.53	8.35	9.89*	9.85*	7.46	9.69	9.62	9.04
Rock	No re-rank	35.03	38.88	42.27	35.03	36.28	31.21	38.98	35.75	40.18	36.57	35.98	40.01	34.10	36.94
	LHRR	33.46	52.52*	47.17*	33.46	53.94*	31.22	41.82	38.68	39.95	38.18	37.31	43.19	34.62	40.43
	RDPAC	33.67	50.23	44.52*	33.67	52.02*	30.56	39.12	38.94	40.01	37.82	39.35	42.21	34.51	39.74
	BFSTREE	34.73	52.08*	45.23*	34.73	51.96	30.87	41.19	40.21	39.64	37.73	38.00	42.69	35.15	40.33
	RFE	33.99	51.00*	44.03*	33.99	46.21	31.32	41.82	35.35	38.51	37.78	35.49	42.72	33.10	38.87
EDev	No re-rank	0.30	0.28	0.29	0.30	0.23	0.36	0.39	0.36	0.39	0.38	0.35	0.39	0.35	0.34
	LHRR	0.31	0.30	0.31	0.30	0.26	0.35	0.39	0.35	0.39	0.38	0.35	0.39	0.35	0.34
	RDPAC	0.31	0.29	0.30	0.30	0.25	0.35	0.39	0.35	0.39	0.38	0.34	0.39	0.34	0.34
	BFSTREE	0.31	0.29	0.31	0.31	0.25	0.36	0.39	0.35	0.39	0.38	0.35	0.39	0.35	0.34
	RFE	0.30	0.29	0.31	0.31	0.26	0.35	0.39	0.35	0.39	0.38	0.35	0.39	0.34	0.34
Yoga	No re-rank	0.54	0.52	0.54	0.54	0.50	0.55*	0.51	0.46	0.52	0.51	0.47	0.52	0.55	0.52
	LHRR	0.54	0.53	0.54	0.54	0.53	0.55*	0.52	0.48	0.53	0.52	0.48	0.53	0.55	0.52
	RDPAC	0.54	0.53	0.54	0.54	0.53	0.56	0.52	0.48	0.53	0.52	0.48	0.53	0.55	0.53
	BFSTREE	0.54	0.53	0.54	0.54	0.53	0.56	0.52	0.48	0.53	0.52	0.48	0.53	0.55	0.53
	RFE	0.54	0.53	0.54	0.54	0.53	0.55*	0.51	0.47	0.53	0.52	0.48	0.53	0.55	0.52
Mean		13.81	15.77	15.55	13.81	16.44	11.11	13.95	12.47	13.63	13.77	13.11	15.81	14.12	

selection is application-dependent. Also, the application of different feature extractors, distance measures, and re-ranking approaches is possible, following the same proposed workflow.

Future work involves investigating aggregation methods for ranking and correlation analysis. The starting point will be the selection of the most promising methods, based on the comparison presented in this work, and aggregation methods associated with re-ranking formulations [31], [34], [58]. Another research direction relies on the use of optimization or learning methods to automatically discover the best combination of descriptors and re-ranking methods, where also automatic parameter tuning or adaptive heuristics for parameter selection would be suitable in this scenario. The Genetic Programming framework investigated in [76] is a promising starting point. Investigating the use of other layers' output of neural networks for feature extraction in GAF, MTF, and RP images seems a promising approach. Also, optimizations regarding each step of the pipeline could also be performed, considering efficiency and scalability concerns. Dimensionality reduction techniques, e.g., UMAP [77], can be employed in the high-dimensional features, not only mitigating the curse of dimensionality but also allowing faster computations regarding the next steps of the pipeline. The employed manifold learning algorithms work with both distance matrices and pre-computed rankings, and the use of indexing techniques has been employed previously in this scenario [78]. Integrating efficient indexing

structures, such as Ball Trees [79] and HNSW [80], would reduce the cost of the nearest neighbor search. Finally, parallelization techniques and GPU-based implementations would also further enhance the efficiency of the pipeline, as demonstrated in [41], [81], and [82]. These optimizations, paired with sliding windows with pre-filtering [83], buffering schemes [84], or asynchronous re-ranking approaches [85] are fundamental to maintain low latency in real time systems. Also, as explored in this work, diverse graph formulations were explored in search tasks [31], [34], [60]. Graph Neural Networks (GNNs) are powerful tools that can capture higher-order structural relationships, complex patterns, and interdependencies between nodes [86]. Their application to time series data has been promising in diverse scenarios [87]. GNNs could enhance our framework by improving data representation, considering both the distances between neighbors and the intrinsic structure of the graph, as well as GNN-based retrieval. Exploring information such as heterophily in graphs could also enhance retrieval results, since many methods assume the homogeneity of a neighborhood [88].

APPENDIX SUPPLEMENTARY MATERIAL

Tables 6, 7, 8, and 9 present the average values of Precision@5 (P@5), Precision@10 (P@10), Recall@5 (R@5), and Recall@10 (R@10), respectively. The same methodology adopted for the computation of Table 4 was utilized.

It is possible to observe that the results between P@5, P@10, R@5, and R@10 are similar to each other, i.e., the best-performing scores are observed for the same combinations of representations and re-ranking methods. The results, however, are different from those presented in Table 4, which suggests that the definition of the metric for determining the best-performing combination is important. For the Beef dataset, the best results were observed for RP-ViT combined with RDPAC (the highest gains are observed from the combination with MTF-ViT and RFE in the first five positions of the ranked list, while for ten positions, Moment with RFE stands). For the CBF dataset, the best results were observed for the Baseline and DFT, considering various re-ranking methods, and for R@5, also DWT-App, Moment, and ROCKET (the highest gains were observed in DWT-Detail with BFSTREE). The best results were RP-ResNet152 for the GunPoint dataset, considering diverse re-ranking methods. In Table 8, we can also observe a highlight in RP-ViT associated with RDPAC (the highest gains were observed in DWT-detail with BFSTREE, also considering RDPAC when analyzing P@5 and R@5). In Table 4, it is possible to observe that RP-ResNet152 performed well. Results on the Rock dataset were similar, considering precision, recall, and mAP values, in which the best results and highest gains were observed for the combination of DWT-Detail and LHRR. On Electric Devices, the best overall results were observed for GAF-ResNet152, and for Recall results, also on RP-ResNet152 and RP-ViT (highest gains observed on DWT-Detail). The best results for the Yoga dataset were observed for the ROCKET descriptor (the highest gains were observed for DWT-Detail and, for R@5, also on RP-ResNet152 and RP-ViT).

Frequently, DWT-Detail is associated with the highest gains. For precision results (Tables 6 and 7), the feature with the highest values, in general, was RP-ViT, while the best re-rank method was LHRR for P@5 and BFSTREE for P@10. For recall (Tables 8 and 9), DWT-Detail also arises as the feature with the highest values. RDPAC led to the best general re-ranking results, as in mAP. For R@5, also positive results were observed for LHRR, and, for R@10, BFSTREE.

REFERENCES

- [1] Z. Zeng, T. Balch, and M. Veloso, "Deep video prediction for time series forecasting," in *Proc. 2nd ACM Int. Conf. AI Finance*, New York, NY, USA, Nov. 2021, pp. 1–7, doi: 10.1145/3490354.3494404.
- [2] A. Falcon, G. D'Agostino, O. Lanz, G. Brajnik, C. Tasso, and G. Serra, "Neural Turing machines for the remaining useful life estimation problem," *Comput. Ind.*, vol. 143, Dec. 2022, Art. no. 103762.
- [3] P. Hewage, A. Behera, M. Trovati, E. Pereira, M. Ghahremani, F. Palmieri, and Y. Liu, "Temporal convolutional neural (TCN) network for an effective weather forecasting using time-series data from the local weather station," *Soft Comput.*, vol. 24, no. 21, pp. 16453–16482, Nov. 2020, doi: 10.1007/s00500-020-04954-0.
- [4] E. F. Cabral, "Using similarity self-join techniques," Ph.D. dissertation, Inst. Math. Comput. Sci. (ICMC), Universidade de São Paulo, São Paulo, Brazil, 2021.
- [5] J. Almeida, D. C. G. Pedronette, B. C. Alberton, L. P. C. Morellato, and R. D. S. Torres, "Unsupervised distance learning for plant species identification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 12, pp. 5325–5338, Dec. 2016.
- [6] S. Pailot-Bonnétat, A. J. L. Harris, S. Calvari, M. De Michele, and L. Gurioli, "Plume height time-series retrieval using shadow in single spatial resolution satellite images," *Remote Sens.*, vol. 12, no. 23, p. 3951, Dec. 2020. [Online]. Available: <https://www.mdpi.com/2072-4292/12/23/3951>
- [7] 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 *Proc. AAAI Conf. Artif. Intell.*, Apr. 2020, vol. 34, no. 2, pp. 1403–1411. [Online]. Available: <https://ojs.aaai.org/index.php/AAAI/article/view/5497>
- [8] J. Y. Campbell and N. G. Mankiw, "Consumption, income, and interest rates: Reinterpreting the time series evidence," *NBER Macroeconomics Annu.*, vol. 4, pp. 185–216, Nov. 1989. [Online]. Available: <https://ideas.repec.org/h/nbr/nberch/10965.html>
- [9] L. Chen, D. Chen, F. Yang, and J. Sun, "A deep multi-task representation learning method for time series classification and retrieval," *Inf. Sci.*, vol. 555, pp. 17–32, May 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0020025520312287>
- [10] D. Song, N. Xia, W. Cheng, H. Chen, and D. Tao, "Deep r -th root of rank supervised joint binary embedding for multivariate time series retrieval," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, New York, NY, USA, Jul. 2018, pp. 2229–2238, doi: 10.1145/3219819.3220108.
- [11] S.-W. Kim, J. Yoon, S. Park, and J.-I. Won, "Shape-based retrieval in time-series databases," *J. Syst. Softw.*, vol. 79, no. 2, pp. 191–203, Feb. 2006. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0164121205000993>
- [12] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, and E. Keogh, "Querying and mining of time series data: Experimental comparison of representations and distance measures," *Proc. VLDB Endowment*, vol. 1, no. 2, pp. 1542–1552, Aug. 2008, doi: 10.14778/1454159.1454226.
- [13] F. Bovolo, B. Demir, and L. Bruzzone, "A cluster-based approach to content based time series retrieval (CBTSR)," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2015, pp. 2793–2796.
- [14] C. Yu, L. Luo, L. L.-H. Chan, T. Raktanmanon, and S. Nutanong, "A fast LSH-based similarity search method for multivariate time series," *Inf. Sci.*, vol. 476, pp. 337–356, Feb. 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0020025518308430>
- [15] E. S. Santos, B. Alberton, L. P. Morellato, and R. da Silva Torres, "Pixelwise time series retrieval in phenological studies," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2019, pp. 6586–6589.
- [16] C. M. Yeh, H. Chen, X. Dai, Y. Zheng, J. Wang, V. Lai, Y. Fan, A. Der, Z. Zhuang, L. Wang, W. Zhang, and J. M. Phillips, "An efficient content-based time series retrieval system," in *Proc. 32nd ACM Int. Conf. Inf. Knowl. Manage.*, New York, NY, USA, 2023, p. 4909, doi: 10.1145/3583780.3614655.
- [17] C.-C.-M. Yeh, H. Chen, X. Dai, Y. Zheng, Y. Fan, V. Lai, J. Wang, A. Der, Z. Zhuang, L. Wang, and W. Zhang, "Temporal treasure hunt: Content-based time series retrieval system for discovering insights," in *Proc. IEEE Int. Conf. Big Data*, Dec. 2023, pp. 1994–1997.
- [18] R. Nawaz, K. H. Cheah, H. Nisar, and V. V. Yap, "Comparison of different feature extraction methods for EEG-based emotion recognition," *Biocybernetics Biomed. Eng.*, vol. 40, no. 3, pp. 910–926, Jul. 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0208521620300553>
- [19] Z. Wang and T. Oates, "Encoding time series as images for visual inspection and classification using tiled convolutional neural networks," in *Proc. 29th Workshops AAAI Conf. Artif. Intell.*, 2015, pp. 40–46.
- [20] P. Senin and S. Malinchik, "SAX-VSM: Interpretable time series classification using SAX and vector space model," in *Proc. IEEE 13th Int. Conf. Data Mining*, Dec. 2013, pp. 1175–1180.
- [21] J. Lines, L. M. Davis, J. Hills, and A. Bagnall, "A shapelet transform for time series classification," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, New York, NY, USA, Aug. 2012, pp. 289–297, doi: 10.1145/2339530.2339579.
- [22] P. Schäfer, "The BOSS is concerned with time series classification in the presence of noise," *Data Mining Knowl. Discovery*, vol. 29, no. 6, pp. 1505–1530, Nov. 2015, doi: 10.1007/s10618-014-0377-7.
- [23] P. Trirat, Y. Shin, J. Kang, Y. Nam, J. Na, M. Bae, J. Kim, B. Kim, and J.-G. Lee, "Universal time-series representation learning: A survey," 2024, *arXiv:2401.03717*.

- [24] N. Arica and F. T. Yarman Vural, "BAS: A perceptual shape descriptor based on the beam angle statistics," *Pattern Recognit. Lett.*, vol. 24, nos. 9–10, pp. 1627–1639, Jun. 2003.
- [25] J.-P. Eckmann, S. O. Kamphorst, and D. Ruelle, "Recurrence plots of dynamical systems," *Europhys. Lett. (EPL)*, vol. 4, no. 9, pp. 973–977, Nov. 1987.
- [26] P. Senin, "Dynamic time warping algorithm review," Dept. Inf. Comput. Sci., Univ. Hawaii Manoa, Honolulu, HI, USA, Tech. Rep. CSDL-08-04, Jan. 2009.
- [27] F. T. Lima and V. M. A. Souza, "A large comparison of normalization methods on time series," *Big Data Res.*, vol. 34, Nov. 2023, Art. no. 100407, doi: [10.1016/j.bdr.2023.100407](https://doi.org/10.1016/j.bdr.2023.100407).
- [28] Y. M. I. Hassan, A. Elkorany, and K. Wassif, "Utilizing social clustering-based regression model for predicting student's GPA," *IEEE Access*, vol. 10, pp. 48948–48963, 2022.
- [29] C. Holder, M. Middlehurst, and A. Bagnall, "A review and evaluation of elastic distance functions for time series clustering," *Knowl. Inf. Syst.*, vol. 66, no. 2, pp. 765–809, Sep. 2023, doi: [10.1007/s10115-023-01952-0](https://doi.org/10.1007/s10115-023-01952-0).
- [30] L. Wei and E. Keogh, "Semi-supervised time series classification," in *Proc. 12th ACM SIGKDD Int. Conf. Knowl. discovery data mining*, Aug. 2006, pp. 748–753.
- [31] D. C. G. Pedronette, L. P. Valem, J. Almeida, and R. D. S. Torres, "Multimedia retrieval through unsupervised hypergraph-based manifold ranking," *IEEE Trans. Image Process.*, vol. 28, no. 12, pp. 5824–5838, Dec. 2019.
- [32] D. C. G. Pedronette, F. M. F. Gonçalves, and I. R. Guilherme, "Unsupervised manifold learning through reciprocal kNN graph and connected components for image retrieval tasks," *Pattern Recognit.*, vol. 75, pp. 161–174, Mar. 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0031320317301978>
- [33] L. P. Valem and D. C. G. Pedronette, "Person re-ID through unsupervised hypergraph rank selection and fusion," *Image Vis. Comput.*, vol. 123, Jul. 2022, Art. no. 104473, doi: [10.1016/j.imavis.2022.104473](https://doi.org/10.1016/j.imavis.2022.104473).
- [34] L. P. Valem, D. C. G. Pedronette, and L. J. Latecki, "Rank flow embedding for unsupervised and semi-supervised manifold learning," *IEEE Trans. Image Process.*, vol. 32, pp. 2811–2826, 2023, doi: [10.1109/TIP.2023.3268868](https://doi.org/10.1109/TIP.2023.3268868).
- [35] M. S. Sarfraz, A. Schumann, A. Eberle, and R. Stiefelhagen, "A pose-sensitive embedding for person re-identification with expanded cross neighborhood re-ranking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 420–429.
- [36] S. Bai, Z. Zhou, J. Wang, X. Bai, L. J. Latecki, and Q. Tian, "Automatic ensemble diffusion for 3D shape and image retrieval," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 88–101, Jan. 2019. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85051049899&doi=10.1109%2FTIP.2018.2863028&partnerID=40&md5=113eef1f03314b559c517cca60041e98>
- [37] A. Iscen, G. Tolias, Y. Avrithis, and O. Chum, "Mining on manifolds: Metric learning without labels," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, Jun. 2018, pp. 7642–7651, doi: [10.1109/CVPR.2018.00797](https://doi.org/10.1109/CVPR.2018.00797).
- [38] R. D. S. Torres, M. Hasegawa, S. Tabbone, J. Almeida, J. A. dos Santos, B. Alberton, and L. P. C. Morellato, "Shape-based time series analysis for remote phenology studies," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2013, pp. 3598–3601.
- [39] R. D. S. Torres and A. X. Falcão, "Content-based image retrieval: Theory and applications," *Revista de Informática Teórica e Aplicada*, vol. 13, pp. 161–185, Mar. 2006.
- [40] D. C. G. Pedronette and R. D. S. Torres, "Exploiting clustering approaches for image re-ranking," *J. Vis. Lang. Comput.*, vol. 22, no. 6, pp. 453–466, Dec. 2011. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1045926X11000632>
- [41] F. Pisani, L. Pascotti Valem, D. C. Guimarães Pedronette, R. D. S. Torres, E. Borin, and M. Breternitz, "A unified model for accelerating unsupervised iterative re-ranking algorithms," *Concurrency Comput., Pract. Exper.*, vol. 32, no. 14, p. 5702, Jul. 2020. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/cpe.5702>
- [42] B. Rozin and D. C. G. Pedronette, "Time series classification using shape features based on angle statistics," in *Proc. Encontro Nacional de Inteligência Artificial e Computacional*, Nov. 2021, pp. 470–481. [Online]. Available: <https://sol.sbc.org.br/index.php/eniac/article/view/18276>
- [43] L. Ye and E. Keogh, "Time series shapelets: A novel technique that allows accurate, interpretable and fast classification," *Data Mining Knowl. Discovery*, vol. 22, nos. 1–2, pp. 149–182, Jan. 2011.
- [44] P. Schäfer and M. Höggqvist, "SFA: A symbolic Fourier approximation and index for similarity search in high dimensional datasets," in *Proc. 15th Int. Conf. Extending Database Technol.*, New York, NY, USA, Mar. 2012, pp. 516–527, doi: [10.1145/2247596.2247656](https://doi.org/10.1145/2247596.2247656).
- [45] P. Chaovalit, A. Gangopadhyay, G. Karabatis, and Z. Chen, "Discrete wavelet transform-based time series analysis and mining," *ACM Comput. Surv.*, vol. 43, no. 2, pp. 1–37, Feb. 2011, doi: [10.1145/1883612.1883613](https://doi.org/10.1145/1883612.1883613).
- [46] A. Dempster, F. Petitjean, and G. I. Webb, "ROCKET: Exceptionally fast and accurate time series classification using random convolutional kernels," *Data Mining Knowl. Discovery*, vol. 34, no. 5, pp. 1454–1495, Jul. 2020, doi: [10.1007/s10618-020-00701-z](https://doi.org/10.1007/s10618-020-00701-z).
- [47] M. Goswami, K. Szafer, A. Choudhry, Y. Cai, S. Li, and A. Dubrawski, "MOMENT: A family of open time-series foundation models," 2024, *arXiv:2402.03885*.
- [48] K.-P. Chan and A. Wai-Chee Fu, "Efficient time series matching by wavelets," in *Proc. 15th Int. Conf. Data Eng.*, 1999, pp. 126–133.
- [49] F. A. Faria, J. Almeida, B. Alberton, L. P. C. Morellato, and R. D. S. Torres, "Fusion of time series representations for plant recognition in phenology studies," *Pattern Recognit. Lett.*, vol. 83, pp. 205–214, Nov. 2016. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S016786551600074X>
- [50] N. Menini, A. E. Almeida, R. Lamparelli, G. Le Maire, J. A. dos Santos, H. Pedrini, M. Hirota, and R. D. S. Torres, "A soft computing framework for image classification based on recurrence plots," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 2, pp. 320–324, Feb. 2019.
- [51] D. Dias, A. Pinto, U. Dias, R. Lamparelli, G. Le Maire, and R. D. S. Torres, "A multirepresentational fusion of time series for pixelwise classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 4399–4409, 2020.
- [52] D. Dias, U. Dias, N. Menini, R. Lamparelli, G. Le Maire, and R. D. S. Torres, "Image-based time series representations for pixelwise eucalyptus region classification: A comparative study," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 8, pp. 1450–1454, Aug. 2020.
- [53] Y. Liu, H. Pu, and D.-W. Sun, "Efficient extraction of deep image features using convolutional neural network (CNN) for applications in detecting and analysing complex food matrices," *Trends Food Sci. Technol.*, vol. 113, pp. 193–204, Jul. 2021.
- [54] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
- [55] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
- [56] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. 9th Int. Conf. Learn. Represent.*, 2021, pp. 1–21.
- [57] J. Kim, K. Shim, J. Kim, and B. Shim, "Vision transformer-based feature extraction for generalized zero-shot learning," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Jun. 2023, pp. 1–5.
- [58] D. C. Guimarães Pedronette, L. Pascotti Valem, and L. J. Latecki, "Efficient rank-based diffusion process with assured convergence," *J. Imag.*, vol. 7, no. 3, p. 49, Mar. 2021. [Online]. Available: <https://www.mdpi.com/2313-433X/7/3/49>
- [59] Q. Zeng and H. Yu, "Group affinity guided deep hypergraph model for person re-identification," *Electron. Lett.*, vol. 55, no. 4, pp. 186–188, Feb. 2019.
- [60] D. C. G. Pedronette, L. P. Valem, and R. D. S. Torres, "A BFS-tree of ranking references for unsupervised manifold learning," *Pattern Recognit.*, vol. 111, Mar. 2021, Art. no. 107666.
- [61] W. Webber, A. Moffat, and J. Zobel, "A similarity measure for indefinite rankings," *ACM Trans. Inf. Syst.*, vol. 28, no. 4, pp. 1–38, Nov. 2010, doi: [10.1145/1852102.1852106](https://doi.org/10.1145/1852102.1852106).
- [62] H. A. Dau, E. Keogh, K. Kamgar, C.-C. M. Yeh, Y. Zhu, S. Gharghabi, C. A. Ratanamahatana, Yanping, B. Hu, N. Begum, A. Bagnall, A. Mueen, G. Batista, and Hexagon-ML. (Oct. 2018). *The Ucr Time Series Classification Archive*. [Online]. Available: https://www.cs.ucr.edu/~eamonn/time_series_data_2018/

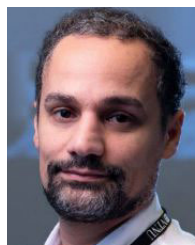
- [63] N. Saito, "Local feature extraction and its applications using a library of bases," Dept. Math., Yale University, New Haven, CT, USA, Tech. Rep. AA19523225, 1994.
- [64] C. Ratanamahatana and E. Keogh, "Everything you know about dynamic time warping is wrong," in *Proc. 3rd Workshop Mining Temporal Sequential Data (TDM)*. Seattle, WA, USA: ACM, 2004. [Online]. Available: <https://www.cs.ucr.edu/~eamonn/TDM04.pdf>
- [65] O. Al-Jowder, E. K. Kemsley, and R. H. Wilson, "Detection of adulteration in cooked meat products by mid-infrared spectroscopy," *J. Agricult. Food Chem.*, vol. 50, no. 6, pp. 1325–1329, Feb. 2002.
- [66] A. M. Baldridge, S. J. Hook, C. I. Grove, and G. Rivera, "The ASTER spectral library version 2.0," *Remote Sens. Environ.*, vol. 113, no. 4, pp. 711–715, Apr. 2009. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0034425708003441>
- [67] A. Bagnall, L. Davis, J. Hills, and J. Lines, "Transformation based ensembles for time series classification," in *Proc. SIAM Int. Conf. Data Mining*, Apr. 2012, pp. 307–318.
- [68] K. Zuva, "Evaluation of information retrieval systems," *Int. J. Comput. Sci. Inf. Technol.*, vol. 4, no. 3, pp. 35–43, 2012.
- [69] J. Faouzi and H. Janati, "Pyts: A Python package for time series classification," *J. Mach. Learn. Res.*, vol. 21, no. 46, pp. 1–6, 2020. [Online]. Available: <http://jmlr.org/papers/v21/19-763.html>
- [70] J. D. Hunter, "Matplotlib: A 2D graphics environment," *Comput. Sci. Eng.*, vol. 9, no. 3, pp. 90–95, 2007.
- [71] J.-W. Bi, H. Li, and Z.-P. Fan, "Tourism demand forecasting with time series imaging: A deep learning model," *Ann. Tourism Res.*, vol. 90, Sep. 2021, Art. no. 103255. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S016073832100133X>
- [72] G. Lee, R. Gommers, F. Waselewski, K. Wohlfahrt, and A. O'Leary, "PyWavelets: A Python package for wavelet analysis," *J. Open Source Softw.*, vol. 4, no. 36, p. 1237, Apr. 2019.
- [73] L. P. Valem and D. C. G. Pedronette, "An unsupervised distance learning framework for multimedia retrieval," in *Proc. ACM Int. Conf. Multimedia Retr.*, New York, NY, USA, Jun. 2017, pp. 107–111, doi: [10.1145/3078971.3079017](https://doi.org/10.1145/3078971.3079017).
- [74] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bull.*, vol. 1, no. 6, pp. 80–83, Dec. 1945. [Online]. Available: <http://www.jstor.org/stable/3001968>
- [75] B. C. Giao and P. C. Vinh, "A framework for similarity search in streaming time series based on spark streaming," *Mobile Netw. Appl.*, vol. 27, no. 5, pp. 2084–2097, Oct. 2022, doi: [10.1007/s11036-022-01988-6](https://doi.org/10.1007/s11036-022-01988-6).
- [76] J. A. V. Muñoz, R. da Silva Torres, and M. A. Gonçalves, "A soft computing approach for learning to aggregate rankings," in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage.*, Melbourne, VIC, Australia, Oct. 2015, pp. 83–92, doi: [10.1145/2806416.2806478](https://doi.org/10.1145/2806416.2806478).
- [77] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform manifold approximation and projection for dimension reduction," 2018, [arXiv:1802.03426](https://arxiv.org/abs/1802.03426).
- [78] D. C. G. Pedronette, J. Almeida, and R. da S. Torres, "A scalable re-ranking method for content-based image retrieval," *Inf. Sci.*, vol. 265, pp. 91–104, May 2014. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0020025113008864>
- [79] S. M. Omohundro, "Five balltree construction algorithms," International Computer Science Institute, Berkeley, CA, USA, Tech. TR-89-063, 1989. [Online]. Available: <https://steveomohundro.com/wp-content/uploads/2009/03/omohundro89fiveballtreeconstructionalgorithms.pdf>
- [80] Y. A. Malkov and D. A. Yashunin, "Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 4, pp. 824–836, Apr. 2020.
- [81] F. Pisani, D. C. G. Pedronette, R. D. S. Torres, and E. Borin, "Contextual spaces re-ranking: Accelerating the re-sort ranked lists step on heterogeneous systems," *Concurrency Comput., Pract. Exper.*, vol. 29, no. 22, Nov. 2017, Art. no. e3962.
- [82] L. P. Valem, D. C. G. Pedronette, R. D. S. Torres, E. Borin, and J. Almeida, "Effective, efficient, and scalable unsupervised distance learning in image retrieval tasks," in *Proc. 5th ACM Int. Conf. Multimedia Retr.*, New York, NY, USA, Jun. 2015, pp. 51–58, doi: [10.1145/2671188.2749336](https://doi.org/10.1145/2671188.2749336).
- [83] A. Ermshaus, P. Schäfer, and U. Leser, "Raising the CLaSS of streaming time series segmentation," *Proc. VLDB Endowment*, vol. 17, no. 8, pp. 1953–1966, Apr. 2024, doi: [10.14778/3659437.3659450](https://doi.org/10.14778/3659437.3659450).
- [84] E. Kim, Y. Cho, and H. Shin, "Cooperative buffering schemes for time-shifted live streaming of distributed appliances," *Appl. Sci.*, vol. 11, no. 23, p. 11527, Dec. 2021. [Online]. Available: <https://www.mdpi.com/2076-3417/11/23/11527>
- [85] A. Jaspal, Q. Dang, and A. Ramineni, "RADAR: Recall augmentation through deferred asynchronous retrieval," 2025, [arXiv:2506.07261](https://arxiv.org/abs/2506.07261).
- [86] M. Li, A. Micheli, Y. G. Wang, S. Pan, P. Lió, G. S. Gnecco, and M. Sanguineti, "Guest editorial: Deep neural networks for graphs: Theory, models, algorithms, and applications," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 4, pp. 4367–4372, Apr. 2024.
- [87] M. Jin, H. Yee Koh, Q. Wen, D. Zambon, C. Alippi, G. I. Webb, I. King, and S. Pan, "A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection," 2023, [arXiv:2307.03759](https://arxiv.org/abs/2307.03759).
- [88] X. Zheng, Y. Wang, Y. Liu, M. Li, M. Zhang, D. Jin, P. S. Yu, and S. Pan, "Graph neural networks for graphs with heterophily: A survey," 2022, [arXiv:2202.07082](https://arxiv.org/abs/2202.07082).



BIONDA ROZIN was born in Rio Claro, São Paulo, Brazil, in 2000. She received the B.Sc. and M.S. degrees in computer science from State University of São Paulo (UNESP), Brazil, in 2022 and 2024, respectively. She is currently a Researcher with UNESP in the project titled 'Software development for automatic recognition of microfossils 1', funded by Petrobras. During the master's program, a teaching internship was completed in the course of algorithm analysis, in 2022. In 2023 and 2025, she taught the course "Operating Systems II" at UNESP. She also completed a Research Internship with Wageningen University and Research, The Netherlands, for two and a half months, in 2023. She received an honorable mention at the workshop of thesis and dissertations of the SIBGRAPI 2025. She investigates the machine learning and information retrieval fields with a focus on clustering, manifold learning, classification, and weakly supervised learning, in the fields of images and time series.



DANIEL CARLOS GUIMARÃES PEDRONETTE received the B.Sc. degree in computer science from São Paulo State University, Brazil, in 2005, and the M.Sc. and Ph.D. degrees in computer science from the University of Campinas, Brazil, in 2008 and 2012, respectively. He is currently an Associate Professor with the Department of Statistics, Applied Mathematics, and Computing, São Paulo State University. His research interests include information retrieval, unsupervised and semi-supervised learning, image analysis, and natural language processing. He is serving as an Associate Editor for *Pattern Recognition* journal.



RICARDO DA SILVA TORRES (Member, IEEE) received the B.Sc. degree in computer engineering and the Ph.D. degree in computer science from the University of Campinas, Brazil, in 2000 and 2004, respectively. He is currently a Professor of data science and artificial intelligence with Wageningen University and Research. He had a position as a Professor of visual computing with the Norwegian University of Science and Technology (NTNU), from 2019 to 2024. He used to hold a position as a Professor with the University of Campinas, from 2005 to 2019. He has been developing multidisciplinary e-science research projects involving multimedia analysis, multimedia retrieval, machine learning, databases, and information visualization. He is the author/co-author of more than 200 papers in refereed journals and conferences, and serves as a PC member for several international and national conferences. He has been serving as an Associate Editor for *Pattern Recognition Letters* and *Pattern Recognition*.

• • •