2025 European Conference on Mobile Robots (ECMR) | 979-8-3315-2705-1/25/$31.00 ©2025 IEEE | DOI: 10.1109/ECMR65884.2025.11163256

Visual Perception of Repetitive Human Motions: Temporal Localization, Grouping and Repetitions Counting

Vasileios Zotos[1,2], Giorgos Karvounas[1,2], Konstantinos Bacharidis[1,2], Antonis Argyros[1,2]

***Abstract*— Given a number of time-series which may exhibit repetitiveness at arbitrary temporal ranges, we identify the following important problems: (1) repetitiveness-based clustering, i.e., time-series clustering on the basis of sharing similar periodicities, (2) localization of temporal repetitiveness, i.e., detection of the start and the end of the repetitive pattern in each time-series cluster, and (3) repetition counting, i.e., estimation of the number of repetitions. Current methods address mainly the third problem (repetition counting). Only a few methods deal with the tasks of temporal localization and repetition-based clustering. In this work, we propose a method that solves all three problems. We also use it for simultaneously clustering, temporally localizing and characterizing multiple co-occurring repetitive human motions. We evaluate our performance against state-of-the-art methods on well-established datasets. We also introduce appropriate variants of existing datasets for assessing performance in individual tasks, as well as a novel dataset that serves as a benchmark for all three tasks. Our experiments demonstrate the effectiveness of our method across all tasks.**

I. INTRODUCTION

Repetitive events and actions appear everywhere in our daily life, including natural phenomena, human behaviors, and mechanical operations. Patterns emerge in the cycles of movement and in the routines that govern both biological and artificial systems. For a robot to function intelligently within such environments, it must be capable of perceiving and interpreting these recurring sequences. Recognizing repetition allows machines to anticipate future occurrences, optimize efficiency, recognize actions and activities [1], and adapt to dynamic surroundings so that they enhance their autonomous capabilities.

Vision offers a passive, non-intrusive means for perceiving repetitive events. Despite its potential, challenges remain in achieving reliable and explainable detection under real-world conditions. Addressing these issues is essential for improving autonomy in collaborative robotic systems. In this work we focus on three related tasks. First, drawing on concepts from dynamical systems theory [3], we introduce P-Group, a novel periodicity-based clustering algorithm to group time series based on the similarity of their repetitiveness. Second, leveraging a wavelet-based time series representation [3], [4], our method performs temporal localization of the repetitive patterns in each of the defined clusters. Finally, our method estimates the number of repetitions in each of them. Our method is the first to natively handle multiple, simultaneous repetitive motions without relying on external tracking modules. Existing techniques that claim to address coexisting repetitions [21] are limited to analyzing individual actors rather than group patterns. Furthermore, many current methods predominantly focus on repetition counting, addressing clustering and temporal localization to a lesser extent [14], [12], [10], [17], [13], [20], [23]. [5] is a pose-level method, which achieved a substantial improvement over all existing video-level methods however it lacks the generality of other methods as it only works for predefined classes of actions.

While our method is applicable to time-series arising in any domain, in this work we focus on the use-case of detecting and analyzing groups of repetitive human motions of similar rythm in videos. To achieve this, a video is converted to a number of time-series of features that represent the temporal behavior of human body joints in a video (see Fig. 1). Our method allows to (1) cluster extracted features from human body joints according to their repetitive motion, (2) localize temporally the repetitive parts of the motion of each joint cluster and, (3) estimate the number of repetitions in each cluster. To the best of our knowledge, this is the first method in the literature that solves simultaneously all these three tasks for multiple co-occurring repetitive human motions.

We evaluate the performance and effectiveness of our method against state-of-the-art methods on well-established datasets and introduce a new dataset supporting all three tasks. Specifically, we demonstrate that our method handles multiple repetitive motion grouping robustly, while achieving performance comparable to state-of-the-art methods on temporal repetitiveness localization and repetition counting.

II. RELATED WORK

Temporal localization of repetitive patterns: The ability to detect recurring patterns in time-series data has been a subject of several research works. Numerous techniques have been developed for identifying periodicities in time-series data, with Fast Fourier Transform and Autocorrelation Function (ACF) methods being the most prevalent approaches. For instance, [9] utilizes the Lomb-Scargle periodogram for periodicity detection, while [7] applies ACF to identify periodic patterns in time-series. However, the periodogram may struggle to detect long periods or handle abrupt changes, while the accuracy of ACF-based approaches can be adversely affected by noise and outliers, impacting peak estimation reliability.

Approaches for temporal localization of repetitive patterns in video sequences mainly consider the self-similarity of frame sequences over time as a localization cue. Specifically, given a frame-wise feature representation extracted using

[1] Computer Science Department, University of Crete, Heraklion, Crete, Greece.
[2] Institute of Computer Science, FORTH, Heraklion, Crete, Greece.

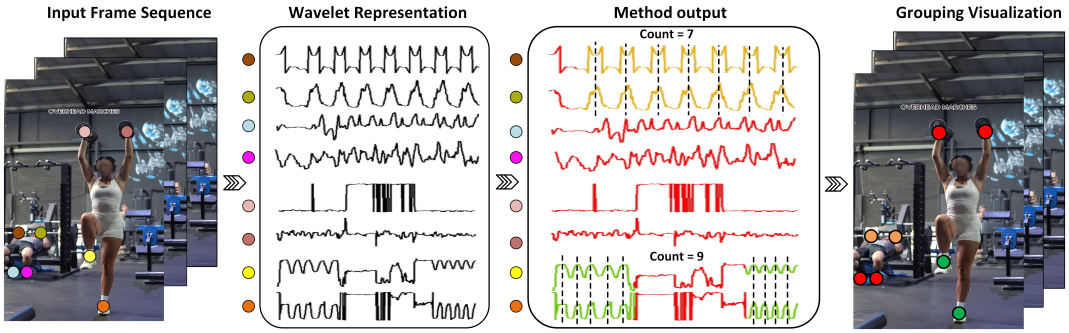

Fig. 1. Given a video, we extract time series of features that quantify the 3D motion of human body joints. Then, the proposed method groups motions on the basis of their repetitiveness, performs temporal segmentation from the non-repetitive parts and counts the number of repetitions. In the method output and grouping visualization sub-figures, red signifies lack of repetitiveness, while green and yellow is used for 2 different clusters of joints, each with different repetitive motion characteristics.

either handcrafted [15] or deep learning [11], [6] features, pairwise frame distances are calculated to construct a temporal self-similarity matrix that serves as a descriptor upon which localization methods identify repetitive segments.

Repetitiveness-based grouping: Spectral methods were successfully applied to cluster time series based on their periodic/repetitiveness properties. Giordano et al. [8] propose a frequency-domain clustering technique designed for stationary time series. Additionally, Puech et al. [16] combines the Discrete Fourier Transform with the autocorrelation function.

In the visual domain, [19] which is currently the state of the art for spatial localization and clustering of repetitive motions in video, introduced a filtering technique that integrates spatial and temporal data to pinpoint periodic motion in video sequences and determine the number of repetitions. To manage non-stationary scenarios, they employ the wavelet transform on the flow field.

Repetition counting in video sequences: To handle non-stationary repetition, [19] adopted the continuous wavelet transform but they still assume that the input is periodic.

Deep learning approaches have also been used to deal with the problem of repetition counting in videos. With the use of TSM as intermediate representation [6], [14] achieve SOTA performance. Methods relying on pose [10], [17], [13] are based on Transformer architectures, with limited ability to effectively capture complex interactions between global body configurations and local joint movements.

To address the problem of non-stationary repetitions, Wang et al. [22] assume uniformity of the spatio-temporal feature representation for repetitive segments, and distinctiveness of the representations of periodic and non-periodic segments. Using optical flow, Li et al. [12], detect periodic patterns by learning a density map to estimate the temporal position of each repetition.

III. Proposed Method

We propose a method that solves simultaneously the problems of repetitiveness-based grouping, temporal repetitiveness localization and repetition counting in time series.

A. Time-series Periodicity Representation

To detect the periodic components of a time series, it is essential to first establish a representation of its time-varying dynamics. For this, we employ wavelets, as they effectively capture both time and frequency information of non-stationary time series, such as videos depicting human actions.

Given the wavelet transform of a time series f, the scalogram summarizes the energy content of the continuous wavelet transform (CWT) of the time series over the entire range of scales. The scalogram of f at a given scale $s > 0$ is given by:

$$S(s) := \langle f, \psi_{u,s}(t) \rangle = \left(\int_{-\infty}^{+\infty} |Wf(u,s)|^2 du \right)^{\frac{1}{2}}, \quad (1)$$

where $Wf(u,s)$ is the continuous wavelet transform of f at time u and scale s.

The scale index i_{scale} of a time series f [3], a wavelet-based tool, provides a measure of the degree of non-periodicity of a signal using the scalogram. Given a scale interval $[s_0, s_1]$, the scale index of f is defined as:

$$i_{scale} = S(s_{min})/S(s_{max}), \quad (2)$$

where $s_{max} \in [s_0, s_1]$ is the smallest scale such that $S(s) \leq S(s_{max})$ for all $s \in [s_0, s_1]$, and $s_{min} \in [s_{max}, 2s_1]$ is the smallest scale such that $S(s_{min}) \leq S(s)$ for all $s \in [s_{max}, 2s_1]$.

In the case of non-stationary signals that exhibit varying periodic behavior, the non-periodic components may not remain constant over time. To tackle this, we incorporate in our approach the windowed scale index, introduced in [4], which is computed similarly to the scale index, but employing the windowed scalogram. The windowed scalogram, is the scalogram limited to a finite time interval $[t - \tau, t + \tau]$.

B. Grouping Time Series with the P-Group Algorithm

We introduce a novel, agglomerative algorithm, which given the scalogram and scale index representation of set of non-stationary signals, is able to identify groups of periodicities. P-Group consists of the following steps.

Initialization: Given a set of signals, we calculate the scalogram and the associated scale index of each signal's

power spectrum. Each signal defines its own cluster. On each iteration, we carry out the calculations outlined below, only to clusters with a scale index below an empirically-determined periodicity threshold, p_{th} (see Sec. IV-C).

Clusters grouping criteria: To determine which pair of clusters should be grouped at each iteration, we add each pair of the available power spectra to create new power spectra. For the newly created power spectra we determine their scale index. We then try to identify a new power spectrum - cluster which satisfies the following criteria:

1) Power spectrum's - cluster's scale index is $i_{scale} < p_{th}$.
2) Power spectrum's - cluster's scale index falls within the range defined by:
   - Lower bound: (1 - σ_p) $\min\{si_1, si_2\}$
   - Upper bound: (1 + σ_p) $\max\{si_1, si_2\}$,

   with $\{si_1, si_2\}$ are the scale indices of the 1st and 2nd contributing clusters and σ_p is is a periodicity deviation parameter (see Sec. IV-C).
3) The power spectrum cluster's scale index is set to be the minimum of the newly calculated scale indices.

Clusters merging: Upon the identification of a new power spectrum - cluster satisfying all criteria, the members of the two contributing clusters are summed to form a single cluster, and the source clusters are discarded.

Termination criterion: The above iterative process is carried out until a state is reached where no additional merging - addition of clusters can be performed.

C. Temporal Localization and Repetitions Counting

Given the clusters of sequences formed by P-Group, along with its calculated scale index, we can temporarily localize the repetitive patterns and estimate the repetitions for each cluster by exploiting the windowed scale index, $wi_{scale,\tau}$. Specifically, for every cluster of signals, we check if its scale index i_{scale} holds that $i_{scale} \leq p_{th}$. If this is not the case, the cluster is classified as non-periodic and its repetition count is set to zero. However, if $i_{scale} \leq p_{th}$ then for each moment in time t, we examine if $wi_{scale,\tau}(t) \leq i_{scale}$. If that is the case, then that time-step is classified as periodic and the peak scale for that time-step is used for repetition counting, otherwise it is not. Specifically, we estimate the repetition count as $\hat{c} = \sum^{n} \frac{dt}{s_n * fc}$, where dt is the time interval between samples, s_n is the scale with the maximum response at each timestep that was classified as periodic and f_c is the Fourier factor, which is a wavelet-specific constant used to convert scales to Fourier periods. For our chosen wavelet, $f_c = 1.07$.

It should be noted that being wavelet-based, our method requires the signal to have more than 3 repetitions. The same holds also for the wavelet-based method of [19].

D. Handling Repetitive Human Actions in Videos

To apply the proposed approach for analyzing repetitive human motions in the visual domain we need to adopt an appropriate representation of the video. In that direction, we chose to extract 3D human skeleton data from video.

To perform 3D pose estimation we use the BlazePose [2], a deep learning approach. For each frame, the network outputs 33 keypoints. Given this set of keypoints we construct a feature vector of time-varying signals calculating for each frame the following quantities:

1. *Pair-wise Joint Distance, D_{Hip}:* This refers to the 3D Euclidean distance between each joint and the hip center.

2. *Joint Angle, $\theta(J_i, J_j)$:* The angle between two connected body joints J_i and J_j, computed with J_j as the reference:

$$\theta(J_i, J_j) = \arctan\left(J_i x - J_j x, J_i y - J_j y\right). \quad (3)$$

The final feature vector is divided into 3 subsets (angles in XY plane, angles in YZ plane, and hip-joint distances). Subsets are examined in this order: XY plane angles, YZ plane angles, and distances. Angle-based features are prioritized over distance-based features due to their superior robustness to various factors of variation, a finding supported by the analysis in [5].

IV. Datasets, Metrics & Settings

A. Dataset Specifications

Existing datasets such as RepCount [10], PERTUBE [15], and UCFRep [23] predominantly target scenarios characterized by a single, dominant repetitive motion in the foreground. Consequently, their utility in contexts involving multiple concurrent repetitive motions is inherently constrained. To address this limitation, we extend these three established datasets for enhanced applicability to vision-based periodicity detection and repetition counting tasks. The datasets encompass a range of sampling rates (15–60 fps) and video durations (6.7–141 seconds), as detailed in Table I.

For each sequence, we compute and annotate two modalities: (i) the 3D Euclidean distances of 33 body joints relative to the hip joint, and (ii) the inter-joint angles between connected keypoints, including the shoulder, elbow, and wrist (bilateral upper limbs), as well as the hip, knee, and ankle (bilateral lower limbs). Each time series is labeled to indicate the presence or absence of visually discernible periodicity, resulting in three derivative datasets: *RepCount-skel*, *PERTUBE-skel*, and *UCFRep-per*.

Furthermore, we introduce a synthetic benchmark, RAL-Synth, designed to support all three target tasks—periodicity detection, repetition localization, and counting—under varying levels of complexity and motion multiplicity. RAL-Synth, along with all supplementary annotations for the modified datasets, will be made publicly available[1] to facilitate standardized evaluation and reproducibility.

RepCount-skel: RepCount-skel is the test set of RepCount dataset with additional human skeleton-level annotation and temporal periodicity annotation. For the temporal periodicity annotation we utilized the existing fine-grained annotation of repetitive actions.

UCFRep-per: UCFRep-per dataset is the validation set of UCFRep [23] dataset. For the temporal periodicity annotation of the UCFRep-per dataset we utilized the existing fine-grained annotation of repetitive actions.

[1] www.ics.forth.gr/cvrl/reptrack/.

Dataset	Number of Videos	Repetitive & Non-Repetitive Motions	Number of Repetitive Motions	Duration (s)			Count		
				Avg	Min	Max	Avg	Min	Max
PERTUBE-skel	34	×	1	27.7	6	77.0	17.4	4.0	68.0
UCFRep	526	×	1	8.2	2.1	33.8	6.7	3.0	54.0
RepCount	1451	×	1	29.4	4.0	88.0	15.9	1.0	141.0
RAL-Synth (**Ours**)	2736	✓	1-2-3	44.1	4.0	176.0	15.9	1.0	141.0

TABLE I

STATISTICS OF VIDEO REPETITION COUNTING DATASETS. RAL-SYNTH EXPANDS DIVERSITY BY INCLUDING COMPLEX SCENARIOS WITH MULTIPLE CONCURRENT REPETITIVE ACTIONS AND COMBINATIONS OF NON-REPETITIVE AND REPETITIVE ACTIONS.

	UCFRep [23]		**RepCount** [10]	
	MAE	**OBOA**	**MAE**	**OBOA**
Transrac [10]	0.640	0.324	0.443	0.291
ESCounts [20]	0.216	0.704	0.245	0.563
RACNet [14]	0.526	0.371	0.444	0.393
MFL [12]	0.388	0.510	0.384	0.386
Repnet [6]	**0.210**	**0.733**	0.331	0.533
IVAC-P2L [22]	0.503	0.420	0.402	0.344
FCA-RAC [13]	0.268	0.470	**0.150**	**0.770**
ME-RAC [17]	0.487	0.460	0.353	0.402
Runia [19]	0.862	0.162	Inf	0.013
Ours	0.769	0.105	0.330	0.283

TABLE II

REPETITION COUNTING PERFORMANCE (MAE↓, OBOA↑).

	Transrac [10]	**Repnet [6]**	**Runia [19]**	**Ours**
MAE	1.848	**0.350**	0.723	0.387
OBOA	0.188	**0.294**	0.147	0.088

TABLE III

REPETITION COUNTING PERFORMANCE (MAE↓, OBOA↑) ON THE PERTUBE-SKEL DATASET.

	Repcount-skel		**PERTUBE-skel**	
	m-Runia	**Ours**	**m-Runia**	**Ours**
Accuracy	0.517	**0.749**	0.445	**0.726**
Precision	0.514	**0.589**	0.516	**0.731**
Recall	0.330	**0.471**	0.267	**0.455**
F1 score	0.333	**0.489**	0.310	**0.533**

TABLE IV

PERFORMANCE ON REPETITIVENESS-BASED GROUPING OF DIFFERENT HUMAN MOTIONS (*m-Runia*: MODIFIED [19]).

PERTUBE-skel: For PERTUBE [15], we superannotate 34 out of the 50 videos of the original dataset. These videos have been supplemented with newly provided skeleton-level annotations. We have retained the temporal periodicity annotations from the original dataset.

RAL-Synth: is a novel synthetic dataset that we introduce as a basis for the evaluation of methods for repetitiveness-based grouping, temporal localization and repetition counting. In Ral-Synth, a synthetic "video" is a collection of 60 1D signals. 30 of these signals are non repetitive (NR) and the other 30 are partly repetitive (PR). There are two types of non-repetitive signals, high-frequency (HF) (Gaussian noise) and low-frequency (LF) (double-log functions). The repetitive part of a PR signal is modeled as a sinusoidal pattern plus Gaussian noise. Each RAL-Synth sequence is associated with a RepCount video and inherits its annotation concerning the number and the start and end-times of each repetition. The PR signals are organized into three classes. For the 1st class we keep the duration of each individual repetitive and non-repetitive segment of each signal the same as in the original videos. For the 2nd and 3rd classes, we increase the duration of each segment by 1.5 and 2 times, respectively. We further create significant duration variation by changing each segment's duration by $\pm 10\%$, following a uniform distribution.

The RAL-Synth is composed of 3 parts, each consisting of synthetic videos with 1, 2 or 3 distinct clusters of repetitive signals. The 30 PR signals of each synthetic video of the first part are from a single class which is randomly chosen. 2) 15 PR signals of each synthetic "video" of the second part are from one class and the other 15 PR signals are from another class. The two classes are chosen randomly. 3) The PR signals of each synthetic "video" of the third part are from all three classes. In summary, to setup the RAL-Synth dataset, we use each of the 152 RepCount videos as a baseline 9 times while producing 2 variants. In one of them, the 30 NR signals and the non-repetitive parts of the 30 PR signals are HF. In the other variant, these are LF. Thus, Ral-Synth consists of $152 \times 9 \times 2 = 2736$ simulated videos.

B. Evaluation Metrics

Repetition Counting: We explore two widely used metrics in the literature [6], [17], [20], [19], (a) the Mean Absolute Error (MAE) and (b) the Off-By-One accuracy (OBOA). The Off-By-One Accuracy (OBOA) metric quantifies the proportion of video sequences for which the count is "nearly perfect". It is a strict classification accuracy measure. Unlike MAE, which offers a continuous-valued, robust measure of overall model precision and is less sensitive to boundary cases than OBOA, OBOA penalizes predictions that fall slightly outside the exact range of correct values (±1), making it highly sensitive to small estimation errors. In scenarios such as activity monitoring or fitness tracking, MAE is the more appropriate metric. It provides a better measure of performance since a low average error is more desirable than a strict adherence to the off-by-one accuracy.

Repetitiveness-Based Grouping: We utilize the Rand Index [18] to evaluate the capability of an algorithm on this task (Table VI). This metric assesses the agreement between the obtained and ground truth partitions, measuring the accuracy of assigning pairs of signals to the same or different clusters based on their repetitiveness. In the case of a single periodic cluster, the task is treated as a binary

	Repcount-skel		UCFRep-per		PERTUBE-skel	
	[6]	Ours	[6]	Ours	[6]	Ours
Accuracy	0.753	**0.761**	0.621	**0.646**	**0.814**	0.762
Precision	**0.888**	0.875	0.867	**0.868**	**0.900**	0.845
Recall	0.754	**0.802**	0.613	**0.646**	**0.838**	0.815
F1 score	0.789	**0.821**	0.672	**0.714**	**0.832**	0.806

TABLE V

PERFORMANCE ON TEMPORAL REPETITIVENESS LOCALIZATION.

classification problem (periodic / non-periodic) and metrics such as accuracy, precision, recall, and F1-score are used for evaluation (Table IV).

Temporal Localization of Repetitiveness: We cast the problem of temporal localization of repetitive patterns as a binary classification problem. Positive samples are instances that are repetitive, while negative are the non-repetitive. Thus, for evaluation we employ standard classification metrics such as accuracy, recall, precision, and F1-score.

C. Implementation Details and Parameters Setting

We tuned the periodicity threshold and deviation utilizing the validation set of RepCount, augmented with human skeleton-level annotation. Specifically, the *Periodicity threshold*, p_{th} value was determined by evaluating thresholds ranging from 0.1 to 0.9, in increments of 0.1, using accuracy, precision, recall, and F1-score. Our results indicated that a threshold of 0.4 leads to the best performance. With this p_{th} value, we evaluated the *Periodicity deviation*, σ_p by running P-Group while increasing the threshold from an initial value of 0.1 to 0.5 with steps of 0.1. The best obtained Rand Index was achieved for $\sigma_p = 0.1$, so we set this as the optimal setting. For computing the windowed scale index we chose a 2-sec time window, which approximates the average cycle duration observed across all datasets. The selected values remained constant for all experiments.

All experiments were conducted on a standard PC. On average, our method requires just one second (CPU) per time series of 29.4 seconds in length.

V. EXPERIMENTAL RESULTS

We present experiments on all three considered problems and in comparison with prominent existing methods. Sample qualitative results are shown in Fig. 2.

A. Repetition Counting

Baseline Methods: We compare our approach against recent deep learning-based video repetition estimation methods [6], [14], [12], [22], [10], [17], [13], [20], [23], as well as the wavelet-based method proposed in [19].

Results: Our method's performance on repetition counting, measured by MAE, is strong but dataset-dependent. On both the RepCount (Table II) and PERTUBE-skel (Table III) datasets, our approach achieves MAE comparable to state-of-the-art deep learning methods like RepNet. It also substantially outperforms the wavelet-based method of [19], whose poor temporal localization leads to failure on signals with sparse repetitions. The primary limitation appears on the UCFRep dataset, where our method's performance is worse. This is an expected outcome, as the data in UCFRep does not satisfy our model's prerequisite (Sec. III-C).

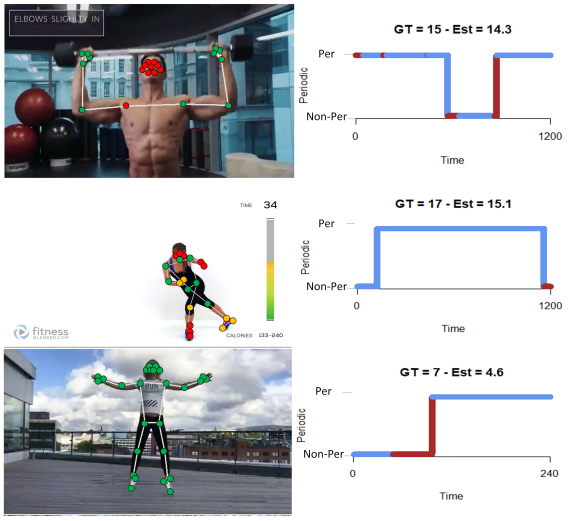

Fig. 2. Qualitative results of our method for identifying repetitive human motions. The 1st column shows color-coded joint clusters (green/orange: repetitive, red: non-repetitive). The 2nd column displays the estimated periodicity over time (high=periodic, low=non-periodic), with blue/brown indicating agreement/disagreement between ground truth (G.T.) and estimations (Est). Repetitions are also indicated. Row 1 (Weight-lifting): Repetitive arm motion identified against non-repetitive face (GT=15, Est=14.3). Row 2 (Lateral hops): Repetitive leg motion identified (GT=17, Est=15.1). Row 3 (Jumping jacks): Repetitive arm and leg motion identified (GT=7, Est=4.6).

However, an analysis of the Off-By-One Accuracy (OBOA) reveals a systemic weakness. Across all datasets, our method achieves low OBOA scores. We attribute this to our method's windowed nature, which fails to capture the exact start and end of a series of repetitions, classifying these boundary segments as non-periodic. The use of a static window size intensifies this problem. A potential solution lies in a dynamic windowing strategy, which we hypothesize would significantly enhance OBOA performance.

B. Repetitiveness-based Grouping of Human Motions

Baseline Methods: We compare the capability of our method to group periodic against non-periodic motions with a modified version of the one proposed in [19]. The employed modification makes [19] applicable to the time series of the RepCount-skel and PERTUBE-skel datasets.

Results: We perform experiments on RepCount-skel and PERTUBE-skel. Since each sample of these datasets contains a single group of repetitive motions, we treat grouping as a binary classification problem that is assessed with standard metrics. Table IV shows that our approach outperforms the modified version of [19] in all reported metrics.

C. Temporal Localization of Repetitive Motions

Baseline Methods: We compare against the deep learning approach of Repnet [6].

Results: Table V shows that our method performs better than [6] on the Repcount-skel and UCFRep-per, but worse when considering PERTUBE-skel. The reduced performance

# Clusters	1 Cluster		2 Clusters		3 Clusters	
NR Type	**HF**	**LF**	**HF**	**LF**	**HF**	**LF**
Rand index	0.656	0.788	0.626	0.739	0.634	0.730
MAE	0.839	0.219	0.560	0.158	0.477	0.158
OBOA	0.500	0.303	0.540	0.328	0.549	0.315
Accuracy	0.808	0.805	0.839	0.805	0.816	0.790
Precision	0.912	0.945	0.918	0.936	0.916	0.937
Recall	0.841	0.823	0.875	0.797	0.858	0.797
F1 score	0.856	0.865	0.880	0.860	0.868	0.848

TABLE VI

ABLATION STUDY ON RAL-SYNTH: THE PERFORMANCE OF THE PROPOSED METHOD IN THE THREE SUBPROBLEMS FOR DIFFERENT NUMBER OF CLUSTERS AND NR SIGNAL TYPES (HF/LF).

of our method can be attributed to Blazepose's difficulty in accurately detecting human body landmarks in certain videos within PERTUBE-skel. Overall, our method despite its multi-task nature performs satisfactorily compared to a method that is specialized for this task.

D. Ablation study on RAL-Synth

We used RAL-Synth to evaluate the influence of the number of repetitive motion clusters and the type of each NR signal (low-frequency - LF, high-frequency - HF) on our method's performance. As shown by the OBOA and MAE scores in Table VI, our method demonstrates consistent performance for repetition counting across all cluster numbers. Performance is better for LF signals. For temporal localization, the accuracy, precision, recall and F1-scores are high, regardless of the type (HF, LF) of NR signals and the number of clusters. Lastly, Rand Index scores confirm the method's repetitiveness-based grouping capability, which is unaffected by variations on the number of clusters. Higher scores are observed for LF signals.

VI. CONCLUSIONS

We proposed a new method that identifies groups of periodic patterns, localizes them in time and estimates their repetition counts. To our knowledge, this is the first to achieve this. Our approach is applicable to various types of temporal data in several domains. In this work, we evaluated it on the problem of repetitive human motion analysis. Experimental results on both well-established and newly-introduced datasets show a competitive performance against methods that are tailored to the individual subproblems. A key direction for future work is to use our method for action recognition and anomaly detection.

ACKNOWLEDGMENTS

This work was carried out within the framework of the Action 'Flagship Research Projects in challenging interdisciplinary sectors with practical applications in Greek industry', implemented through the National Recovery and Resilience Plan Greece 2.0 and funded by the European Union – NextGenerationEU (project code: TAEDR-0535864). This work was also co-funded and supported by the European Union (EU - HE Magician – Grant Agreement 101120731).

REFERENCES

[1] Konstantinos Bacharidis and Antonis Argyros. Repetition-aware image sequence sampling for recognizing repetitive human actions. In *ICCV Workshops*, pages 1878–1887, October 2023.

[2] Valentin Bazarevsky, Ivan Grishchenko, Karthik Raveendran, Tyler Lixuan Zhu, Fan Zhang, and Matthias Grundmann. Blazepose: On-device real-time body pose tracking. *ArXiv*, abs/2006.10204, 2020.

[3] Rafael Benítez, Vicente J Bolós, and ME Ramírez. A wavelet-based tool for studying non-periodicity. *Computers & Mathematics with Applications*, 60(3):634–641, 2010.

[4] Vicente J Bolós, Rafael Benítez, and Román Ferrer. A new wavelet tool to quantify non-periodicity of non-stationary economic time series. *Mathematics*, 8(5):844, 2020.

[5] Haodong Chen, Ming C Leu, Md Moniruzzaman, Zhaozheng Yin, and Solmaz Hajmohammadi. Advancements in repetitive action counting: Joint-based poserac model with improved performance. *arXiv preprint arXiv:2308.08632*, 2023.

[6] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. Counting out time: Class agnostic video repetition counting in the wild. In *CVPR*, pages 10387–10396, 2020.

[7] Mohamed G Elfeky, Walid G Aref, and Ahmed K Elmagarmid. Periodicity detection in time series databases. *IEEE Transactions on Knowledge and Data Engineering*, 17(7):875–887, 2005.

[8] Francesco Giordano, Michele La Rocca, and Maria Lucia Parrella. Clustering complex time-series databases by using periodic components. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 10(2):89–106, 2017.

[9] Earl F Glynn, Jie Chen, and Arcady R Mushegian. Detecting periodic patterns in unevenly spaced gene expression time series using lomb–scargle periodograms. *Bioinformatics*, 22(3):310–316, 2006.

[10] Huazhang Hu, Sixun Dong, Yiqun Zhao, Dongze Lian, Zhengxin Li, and Shenghua Gao. Transrac: Encoding multi-scale temporal correlation with transformers for repetitive action counting. In *IEEE CVPR*, pages 19013–19022, 2022.

[11] Giorgos Karvounas, Iason Oikonomidis, and Antonis Argyros. Reactnet: Temporal localization of repetitive activities in real-world videos. *arXiv preprint arXiv:1910.06096*, 2019.

[12] Xinjie Li and Huijuan Xu. Repetitive action counting with motion feature learning. In *IEEE WACV*, pages 6499–6508, 2024.

[13] Jiada Lu, WeiWei Zhou, Xiang Qian, Dongze Lian, Yanyu Xu, Weifeng Wang, Lina Cao, and Shenghua Gao. Fca-rac: First cycle annotated repetitive action counting. *arXiv preprint arXiv:2406.12178*, 2024.

[14] Yanan Luo, Jinhui Yi, Yazan Abu Farha, Moritz Wolter, and Juergen Gall. Rethinking temporal self-similarity for repetitive action counting. *arXiv:2407.09431*, 2024.

[15] Costas Panagiotakis, Giorgos Karvounas, and Antonis Argyros. Unsupervised detection of periodic segments in videos. In *ICIP*, pages 923–927. IEEE, 2018.

[16] Tom Puech, Matthieu Boussard, Anthony D'Amato, and Gaëtan Millerand. A fully automated periodicity detection in time series. In *Advanced Analytics and Learning on Temporal Data: 4th ECML PKDD Workshop, AALTD 2019*, pages 43–54. Springer, 2020.

[17] Yicheng Qiu, Li Niu, and Feng Sha. Multipath 3d-conv encoder and temporal-sequence decision for repetitive-action counting. *Expert Systems with Applications*, 249:123760, 2024.

[18] William M Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850, 1971.

[19] Tom F. H. Runia, Cees G. M. Snoek, and Arnold W. M. Smeulders. Repetition estimation. *International Journal of Computer Vision*, 127:1361 – 1383, 2018.

[20] Saptarshi Sinha, Alexandros Stergiou, and Dima Damen. Every shot counts: Using exemplars for repetition counting in videos. *arXiv preprint arXiv:2403.18074*, 2024.

[21] Yin Tang, Wei Luo, Jinrui Zhang, Wei Huang, Ruihai Jing, and Deyu Zhang. Multicounter: Multiple action agnostic repetition counting in untrimmed videos. *ArXiv*, abs/2409.04035, 2024.

[22] Hang Wang, Zhi-Qi Cheng, Youtian Du, and Lei Zhang. Ivac-p2l: Leveraging irregular repetition priors for improving video action counting. *CoRR*, 2024.

[23] Huaidong Zhang, Xuemiao Xu, Guoqiang Han, and Shengfeng He. Context-aware and scale-insensitive temporal repetition counting. In *IEEE CVPR*, pages 670–678, 2020.