A Multi-view Extension for Change Point Detection via Time-invariant Representations

Zhenxiang Cao
Dept. of Electrical Engineering, STADIUS Center
KU Leuven
Leuven, Belgium
zhenxiang.cao@esat.kuleuven.be

Nick Seeuws
Dept. of Electrical Engineering, STADIUS Center
KU Leuven
Leuven, Belgium
nick.seeuws@esat.kuleuven.be

Maarten De Vos
Dept. of Electrical Engineering, STADIUS Center
Dept. of Development and Regeneration
KU Leuven
Leuven, Belgium
maarten.devos@kuleuven.be

Alexander Bertrand
Dept. of Electrical Engineering, STADIUS Center
KU Leuven
Leuven, Belgium
alexander.bertrand@kuleuven.be

***Abstract*—Change point detection (CPD) aims at detecting abrupt changes in sequential data. In the past few years, distribution-free CPD models have become a popular alternative for CPD algorithms based on traditional statistical tests. They usually offer superior detection performance and exhibit better generalization capabilities across a spectrum of simulated and real-life datasets. However, many existing CPD approaches either rely solely on information from time domain or frequency domain or balance contributions from different domains through intricate post-processing procedures. Both of these scenarios overlook important realities: 1) Different types of change points can occur within one single recording. 2) Different types of change points often manifest themselves differently in different domains. In response to these challenges, we introduce a multi-view extension of the so-called time-invariant representation (TIRE) CPD model. This multi-view TIRE model possesses the ability to adaptively select piece-wise stationary information from both the time and frequency domains, which are supposed to be helpful for CPD task. When compared to existing baselines, the new model consistently achieves superior or comparable performance across a diverse set of benchmark datasets.**

***Index Terms*—Change point detection, distribution-free, multi-view model**

I. Introduction

Change point detection (CPD) refers to the problem of identifying abrupt changes in observed statistical variables, which is a crucial pre-processing technique in various domains related to time series analysis, including signal processing [1], finance [2], climate science [3], and biomedical informatics [4].

This research received funding from the Flemish Government (AI Research Program) and from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (No.802895). The authors acknowledge the financial support of the FWO (Research Foundation Flanders) for project G0D8321N and G0C9623N. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the granting authorities. Neither the European Union nor the granting authorities can be held responsible for them.

In recent years, a notable shift in CPD research has emerged: a majority of new approaches are grounded in distribution-free models. These models address a key limitation of traditional CPD methods, which heavily depend on predefined parametric models. Among these novel approaches, autoencoder-based structures stand out for their ability to tackle the CPD problem in an inherently unsupervised setting.

For instance, the Autoencoder-Based Breakpoints Detection (ABD) model [5] maps consecutive time windows into latent space features and identifies change points (CPs) by monitoring dissimilarities among these features. Another notable approach, the Time-Invariant Representation (TIRE) [6], [7], divides the autoencoder latent space into time-invariant (TI) and time-variant (TV) feature spaces. The TI features of neighboring windows are encouraged to be similar in the absence of a change point, while a strong dissimilarity is observed when crossing a change point. Different types of CPs are often more readily detectable in specific domains, either the time domain (TD) or the frequency domain (FD). Therefore, to ensure the proposed model works effectively on various datasets, the TIRE model is usually trained independently in both TD and FD, after which the contributions from both domains are fused via a post-processing method [6].

Nevertheless, while the TIRE model achieves state-of-the-art results on diverse simulated and real-life datasets [6], [7], two significant limitations persist: 1) These models necessitate separate training in both TD and FD, which doubles the amount of training time. 2) The data-driven post-processing step proposed in [6] acts as a compromise between information from both domains, often resulting in a decrease in detection performance compared to only utilizing the most informative of both domains.

To address these concerns, we introduce a novel multi-view autoencoder structure capable of accepting both a TD and FD representation of the same signal segment. Our comparative

results against other baseline models illustrate the advantages offered by this new multi-view model, particularly when dealing with datasets containing different types of CPs. We summarize our primary contributions as follows:

- We propose a multi-view autoencoder capable of adaptively preserving CP-relevant information within the TI features extracted from both TD and FD.
- We extend the diamond loss function of [7] for TIRE models to incorporate this new multi-view structure. We will demonstrate that the extracted TI features exhibit a high degree of consistency between consecutive CPs. Hence, CPs can be readily identified by tracking the dissimilarity among the TI features.

II. Brief Introduction to TIRE and Diamond Loss

The TIRE model, as introduced in [6], utilizes an autoencoder that takes windows of a signal as an input, where each window is either represented in the original TD or as the magnitude of its discrete Fourier transform (DFT). The model's latent space is split into two disjoint latent spaces: the TI feature space and the TV feature space. The core concept behind the TIRE structure is as follows: TI features should locate compactly across neighboring windows if no CP is present, while TV features contain window-dependent information necessary for reconstructing the details within the input windows. The detection of CPs relies solely on dissimilarities between extracted TI features.

To ensure that TI features distribute compactly within the same segment, the original TIRE model [6] uses a time-invariant loss, in addition to the reconstruction loss, to supervise the training process. However, this new term introduces a trade-off hyperparameter to balance the two losses. Moreover, information leakage often occurs between the TI and TV features obtained from the TIRE model. This leakage can result in a decline in the final detection performance.

To address these issues, a new loss function was introduced in [7], referred to as the diamond loss, which replaces the weighted sum of the two losses in the original TIRE model. The diamond TIRE model re-couples the TI ($\mathbf{f}^{ti}$) and TV ($\mathbf{f}^{tv}$) features from consecutive windows ($\mathbf{w}_{t-1}$ and $\mathbf{w}_t$) in a single reconstruction loss:

$$\mathcal{L}^{dia} = \sum_t (||\mathbf{w}_t - \widehat{\mathbf{w}}_t||_2^2 + ||\mathbf{w}_{t-1} - \widetilde{\mathbf{w}}_{t-1}||_2^2), \qquad (1)$$

where $\widetilde{\mathbf{w}}_{t-1}$ and $\widehat{\mathbf{w}}_t$ denote the reconstructed windows by re-coupling $\mathbf{f}^{ti}$ and $\mathbf{f}^{tv}$ at time samples $t-1$ and t. The idea behind the diamond loss is that if no CP exists between consecutive windows, the TI features extracted from a specific window can be used for the reconstruction of the next window due to their time-invariant property. The data flow of the diamond TIRE model is illustrated in Fig.1.

To handle situations where it is unclear whether CPs manifest themselves mostly in the TD or the FD, both the original TIRE model and the diamond TIRE model employ identical post-processing procedures. Models with the same architecture are trained independently for both domains (TD and FD), after

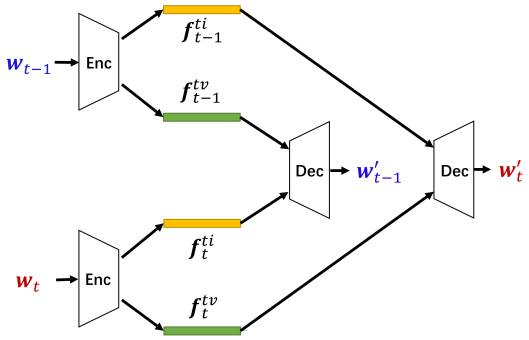

Fig. 1. The data flow in the TIRE model with diamond loss.

which a heuristic post-processing strategy combines the TI features from both models ensuring a balanced contribution from both domains. The concatenated TI feature vector is defined as:

$$\mathbf{f}^{ti,both} = [\alpha \cdot (\mathbf{f}^{ti,TD})^T, \beta \cdot (\mathbf{f}^{ti,FD})^T]^T, \qquad (2)$$

where $\mathbf{f}^{ti,TD}$ and $\mathbf{f}^{ti,FD}$ denote the TI features of the TD and FD model, respectively, and where the values of α and β are determined in a data-driven fashion based on a heuristic rule [6]. Change points are then detected by searching for peaks in the dissimilarities between the feature vectors (2) of subsequent time points (where α and β weigh the influence of each domain in this detection process). While this merging strategy often yields superior detection results compared to other existing CPD methods, it is more akin to a compromise between information from the TD and the FD rather than an elegant, adaptive solution for selecting relevant domain information. Consequently, the detection performance obtained from the merged TI features is typically suboptimal when compared to the (diamond) TIRE model trained exclusively in the appropriate domain. A more significant drawback of this post-processing arises when dealing with time series containing various types of CPs, where some are more prominent in the TD and others in the FD. In such complex scenarios, this data-driven merging strategy loses its effectiveness.

III. Proposed Method

We employ the same pre-processing as described in [6] and [7] to acquire time windows in both the TD and the FD. The choice of window size, denoted as N, is determined based on the expected minimum time between consecutive CPs [6].

As discussed in the previous section, two sequences of TI features (one from TD and the other one from FD) will be extracted by the TIRE model. To address the limitations arising from the heuristic post-processing step (2), we propose a novel multi-view autoencoder, termed 'multi-view TIRE'. This model is designed to process input windows from both TD and FD simultaneously, retaining only the piece-wise stable information from either domain in the TI features. These TI features represent time-invariant information between consecutive windows, such that large shifts in this feature

space can be attributed to CPs. The architecture of the multi-view TIRE is illustrated in Fig.2. In this section, we only conceptually explain the multi-view TIRE model, while concrete implementation details can be found in Section IV-C.

As shown in Fig.2, the multi-view TIRE model consists of three distinct phases. Initially, the input time windows from the time domain ($\mathbf{w}_t^{TD}$) and the frequency domain ($\mathbf{w}_t^{FD}$) are processed independently prior to their integration. Here t is a time index. Using convolutional neural network (CNN)-based encoders, features $\mathbf{f}_t^{TD}$ and $\mathbf{f}_t^{FD}$ are extracted from $\mathbf{w}_t^{TD}$ and $\mathbf{w}_t^{FD}$, respectively. These features are then concatenated to form a combined feature $\mathbf{f}_t$ before proceeding to the subsequent phase. This approach of processing $\mathbf{w}_t^{TD}$ and $\mathbf{w}_t^{FD}$ in parallel pipelines, as opposed to merging TD and FD information from the outset, ensures that the relevant features for reconstruction of each domain are separately extracted, after which the TV versus TI information from each representation can be separated in the second stage. In the second phase, the TI features can either be extracted from both domains, or only from a single domain in case the other domain does not carry TI information. To this end, the model integrates information from both domains by projecting it onto a joint time-frequency feature space, which is split into TI and TV spaces. The split is performed by two parallel fully-connected layers, which extract the TI features ($\mathbf{f}_t^{TI}$) and TV features ($\mathbf{f}_t^{TV}$) from $\mathbf{f}_t$. The separation into TI and TV features is encouraged by the diamond loss, in which the reconstruction of the window at time t is defined by the TI features at time $t-1$. To this end, the TI features of the window at time t are concatenated with the TV features of the window at time $t-1$ (and vice versa), resulting in the concatenated vectors $\widetilde{\mathbf{f}}_{t-1}$ and $\widehat{\mathbf{f}}_t$. This mechanism enables the multi-view TIRE model to encode piece-wise stationary information into TI features, facilitating the precise localization of CPs, while relegating non-critical information to TV features, regardless of its domain origin. Finally, the model maps the combined features back into their original domains again using two parallel fully-connected layers and employs CNN-based decoders to reconstruct the input windows in TD ($\widehat{\mathbf{w}}_t^{TD}$ and $\widetilde{\mathbf{w}}_{t-1}^{TD}$) and FD ($\widehat{\mathbf{w}}_T^{FD}$ and $\widetilde{\mathbf{w}}_{t-1}^{FD}$). The training process of the multi-view TIRE model is supervised using the diamond loss, defined as follows:

$$\mathcal{L} = \sum_t (||\mathbf{w}_t^{TD} - \widehat{\mathbf{w}}_t^{TD}||_2^2 + ||\mathbf{w}_t^{FD} - \widehat{\mathbf{w}}_t^{FD}||_2^2 + ||\mathbf{w}_{t-1}^{TD} - \widetilde{\mathbf{w}}_{t-1}^{TD}||_2^2 + ||\mathbf{w}_{t-1}^{FD} - \widetilde{\mathbf{w}}_{t-1}^{FD}||_2^2). \quad (3)$$

After collecting the sequence of $\{\mathbf{f}_t^{TI}\}_\mathbf{t}$, we applied the same moving average filter as in [6] and [7] to smooth the dissimilarity measure of the TI features and to detect the CPs.

IV. Experiment

A. Benchmark datasets

We assess the performance of the multi-view TIRE model alongside other baseline models across eight simulated datasets and three real-life datasets.

1) Simulated datasets: In our implementation, we generate two groups of simulated datasets. For each dataset, ten independent recordings are produced to ensure generality. The first group is created using the auto-regressive model $s(t) = a_1 s(t-1) + a_2 s(t-2) + \epsilon_t$, as adopted from [6]–[8], where the error term ϵ_t follows a Gaussian distribution $\epsilon_t \sim \mathcal{N}(\mu_t, \sigma_t^2)$. The initial conditions are set as $s(1) = s(2) = 0$, with parameters $a_1 = 0.6$, $a_2 = -0.5$, $\mu_t = 0$, and $\sigma_t = 1.5$, unless specified otherwise. Within this group, each recording is designed to contain only one specific type of CP, characterized as follows:

- **Jumping Mean (JM)**: CPs are introduced by altering the value of μ_t.
- **Scaling Variance (SV)**: CPs are created by adjusting the value of σ_t.
- **Changing Coefficients (CC)**: CPs arise by alternating the value of a_1 between two independent uniform distributions, while a_2 is kept constant at 0.
- **Gaussian Mixture (GM)**: Time samples of $s(t)$ are drawn alternately from two distinct Gaussian mixtures before and after each CP.

In order to analyze how well our method is able to cope with datasets containing different types of change points, we create a second group of simulated datasets characterized by the presence of two different types of CPs alternating within each recording. The naming convention for these datasets combines the acronyms of the CPs they contain. For example, 'JM-SV' denotes recordings that contain CPs in both mean and variance. According to findings in [6], [7], CPs related to JM and GM are more readily detected in the TD, whereas SV and CC CPs are more pronounced in the FD. This deliberate combination of CP types, which exhibit different characteristics in each domain, aims to recreate a scenario where the post-processing procedure described in (2) might lose its effectiveness. Consequently, the second group comprises four datasets: **JM-SV**, **JM-CC**, **GM-SV**, and **GM-CC**, each designed to simulate complex detection environments.

2) Real-life datasets: Beyond the simulated datasets described earlier, we also employ three real-life datasets for our evaluation, which are well-known benchmark datasets for CPD evaluation, including **Honeybee Dance** [9], **Well log** [10], and **HASC-2011** [11].

B. Evaluation metrics and baseline models

Following [12]–[14], we employ the f1-score as the metric to evaluate the CPD performance. We adopt the same criteria as described in [6] to ascertain whether a detected alarm constitutes an actual CP.

We employ the CNN-based diamond TIRE model from [7] as our baseline, which was shown to outperform the simpler multi-layer perceptron TIRE model from [6]. This baseline is designated as 'TIRE (post)', reflecting its utilization of the post-processing procedure in (2) to merge information from both domains. Furthermore, we introduce an additional baseline model that employs the same framework but differs in its approach to domain integration. By directly inputting

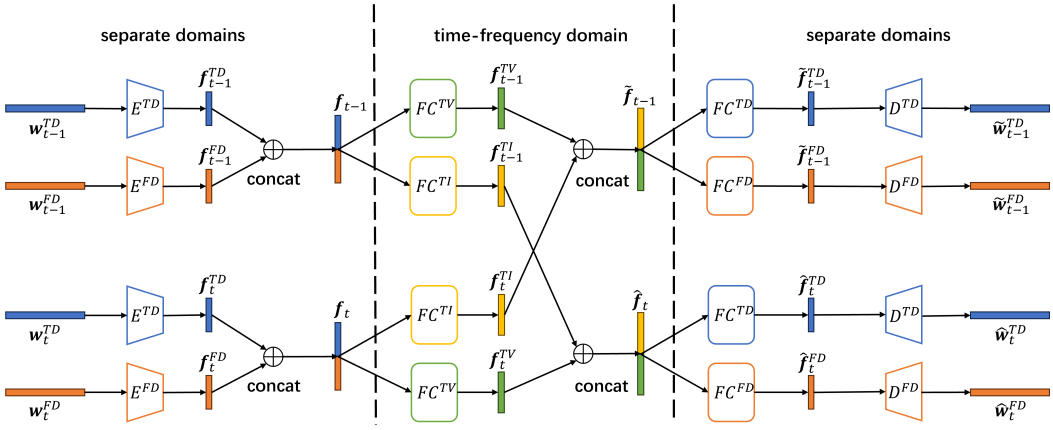

Fig. 2. The architecture of multi-view TIRE model. E stands for a CNN-based encoder, D for CNN-based decoder, and FC for a fully-connected layer.

the concatenated results of time windows from the TD and FD, this model, termed 'TIRE (concat)', combines information from both domains solely through its architecture, eliminating the need for subsequent post-processing. Unlike the multi-view TIRE model, which initially extracts features from both domains separately, the 'TIRE (concat)' model combines this information from the outset.

C. Experiment settings

Following [6], [7], we define the window size N, which determines the time resolution for detecting change points, as $N = 40$ for all simulated datasets, $N = 16$ for the Honeybee Dance dataset, $N = 100$ for the Well Log dataset, and $N = 280$ for the HASC-2011 dataset. A change point (CP) is considered accurately detected if it is identified within a window length N from the actual ground truth CP.

An open-source implementation of our multi-view architecture can be found in an online repository [15], and we only provide a high-level overview here. We utilize 1-dimensional convolutional layers to construct the CNN encoders and decoders. Initially, the input windows from distinct domains are transformed into 4-dimensional features. Subsequently, these features are mapped into $\mathbf{f}^{TI}$ and $\mathbf{f}^{TV}$ through two fully-connected layers, each feature vector having a length of 5. The concatenated $\mathbf{f}^{TI}$ and $\mathbf{f}^{TV}$ features are then processed by another pair of fully-connected layers to regenerate domain-specific features for reconstructing the input windows (again of dimension 4). These are then decoded to reconstruct the original input windows by using a CNN decoder architecture. The optimization process employs an Adam optimizer with a learning rate of 0.001. To mitigate randomness due to initialization and input mini-batch shuffling, we present the average and standard deviation from 10 distinct runs.

D. Results

As illustrated in Fig. 3, the multi-view TIRE model consistently delivers superior or at least comparable detection performance compared to the baseline models. A significant finding is that the multi-view TIRE model not only outperforms the two baseline models—TIRE (post) and TIRE (concat), which comprehensively incorporate information from both domains—but also exceeds the performance of TIRE models trained exclusively with data from a single domain. This underscores the multi-view TIRE model's effective utilization of both time and frequency domain information. This advantage is particularly evident in the results obtained from the second group of simulated datasets, which include CPs that are distinctly observable within different domains. Models trained on data from only one domain struggle with these datasets, whereas the multi-view model demonstrates significantly better performance.

Similar to [7], we perform a representation analysis to examine the TI features extracted by our model. We calculate the distance matrices for the TI features at different time points, where the entry at position (i, j) represents the Euclidean distance between $\mathbf{f}_{t_i}^{ti}$ and $\mathbf{f}_{t_j}^{ti}$. Ideally, this distance should approach zero if no CP exists between t_i and t_j. The emergence of a checkerboard pattern, which signifies that TI features are consistent within segments sharing identical ground truth statistics despite temporal separation, is due to the alternating ground truth statistics across CPs. As depicted in Fig. 4, the distinct tiny dark blocks in the distance matrix in the JM-SV dataset are only observable via the TI features extracted by the multi-view TIRE model. In the Honeybee Dance dataset, while models trained only on frequency domain windows can delineate clear segment boundaries, the post-processing merging strategy dilutes these distinctions by incorporating time domain data. In contrast, the multi-view TIRE model maintains obvious segment boundaries by effectively integrating data from both domains.

V. Conclusion

We have introduced a novel multi-view CPD framework that effectively utilizes information from both the time and frequency domains. Our model selectively retains piece-wise stationary information from both domains within the TI features, thereby enhancing detection performance. Compared to other baseline models, our approach not only achieves superior accuracy but also obviates the necessity for (heuristic)

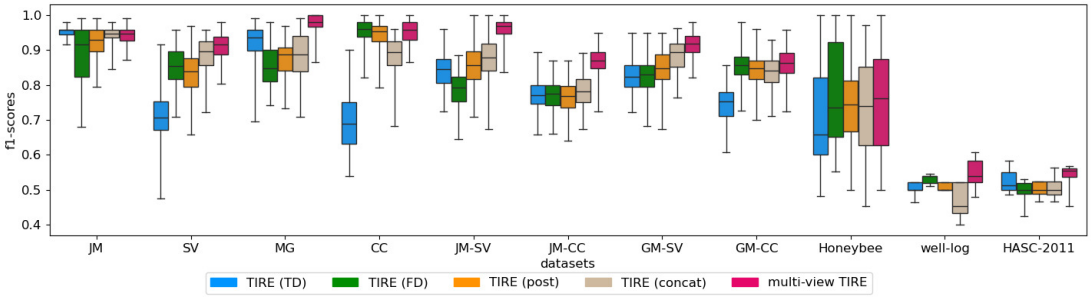

Fig. 3. Evaluation results of multi-view TIRE model and baseline models on all evaluation datasets.

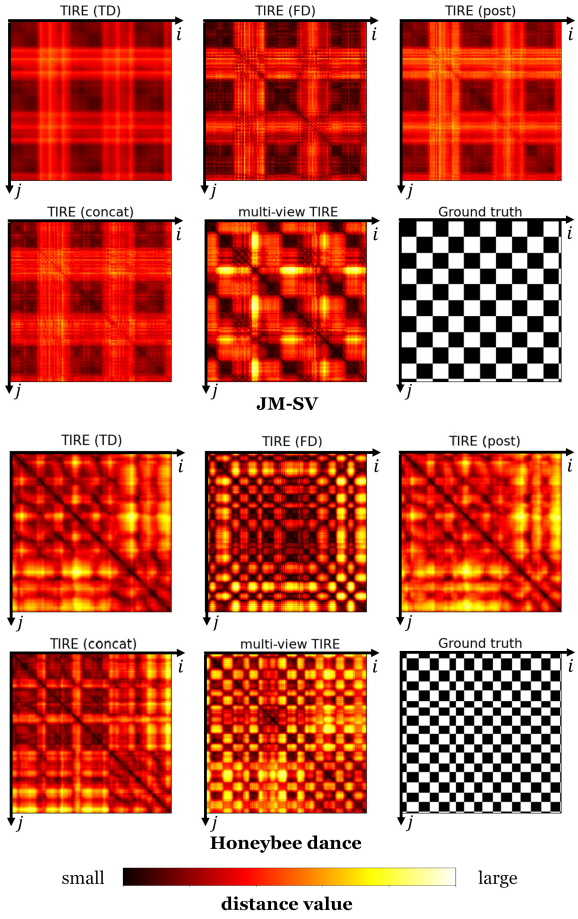

Fig. 4. Visualization of distance matrices of TI features in the form of heatmaps. The entry at position (i, j) represents the distance between TI features $\mathbf{f}_{t_i}^{ti}$ and $\mathbf{f}_{t_j}^{ti}$ at time steps t_i and t_j. The locations of real CPs correspond to the boundaries of the dark blocks along the diagonal in the ground truth sub-figures.

post-processing. This advantage is particularly pronounced in complex datasets featuring multiple types of CPs, where our model demonstrates its robustness and efficiency.

ACKNOWLEDGEMENT

We utilized the Chat Generative Pre-trained Transformer (ChatGPT), a large language model-based chatbot, for proof-reading our manuscript and condensing its content to meet the page limitations. No text was generated by ChatGPT from scratch.

REFERENCES

[1] A. Owrang, M. Malek-Mohammadi, A. Proutiere, and M. Jansson, "Consistent change point detection for piecewise constant signals with normalized fused lasso," *IEEE Signal Processing Letters*, vol. 24, no. 6, pp. 799–803, 2017.

[2] M. Lavielle and G. Teyssiere, "Adaptive detection of multiple change-points in asset price volatility," in *Long memory in economics*. Springer, 2007, pp. 129–156.

[3] J. Reeves, J. Chen, X. L. Wang, R. Lund, and Q. Q. Lu, "A review and comparison of changepoint detection techniques for climate data," *Journal of applied meteorology and climatology*, vol. 46, no. 6, pp. 900–915, 2007.

[4] P. Yang, G. Dumont, and J. M. Ansermino, "Adaptive change detection in heart rate trend monitoring in anesthetized children," *IEEE transactions on biomedical engineering*, vol. 53, no. 11, pp. 2211–2219, 2006.

[5] W.-H. Lee, J. Ortiz, B. Ko, and R. Lee, "Time series segmentation through automatic feature learning," *arXiv preprint arXiv:1801.05394*, 2018.

[6] T. De Ryck, M. De Vos, and A. Bertrand, "Change point detection in time series data using autoencoders with a time-invariant representation," *IEEE Transactions on Signal Processing*, vol. 69, pp. 3513–3524, 2021.

[7] Z. Cao, N. Seeuws, M. De Vos, and A. Bertrand, "A novel loss for change point detection models with time-invariant representations," *IEEE Signal Processing Letters*, vol. 30, pp. 1737–1741, 2023.

[8] S. Liu, M. Yamada, N. Collier, and M. Sugiyama, "Change-point detection in time-series data by relative density-ratio estimation," *Neural Networks*, vol. 43, pp. 72–83, 2013.

[9] S. Oh, J. Rehg, T. Balch, and F. Dellaert, "Learning and inferring motion patterns using parametric segmental switching linear dynamic systems," *International Journal of Computer Vision*, vol. 77, 05 2008.

[10] J. O. Ruanaidh, W. J. Fitzgerald, and K. J. Pope, "Recursive bayesian location of a discontinuity in time series," in *Proceedings of ICASSP'94. IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 4. IEEE, 1994, pp. IV–513.

[11] N. Kawaguchi, Y. Yang, T. Yang, N. Ogawa, Y. Iwasaki, K. Kaji, T. Terada, K. Murao, S. Inoue, Y. Kawahara *et al.*, "Hasc2011corpus: towards the common ground of human activity recognition," in *Proceedings of the 13th international conference on Ubiquitous computing*, 2011, pp. 571–572.

[12] Z. Cao, N. Seeuws, M. D. Vos, and A. Bertrand, "A semi-supervised interactive algorithm for change point detection," *Data Mining and Knowledge Discovery*, pp. 1–29, 2023.

[13] S. Deldari, D. V. Smith, H. Xue, and F. D. Salim, "Time series change point detection with self-supervised contrastive predictive coding," in *Proceedings of the Web Conference 2021*, 2021, pp. 3124–3135.

[14] G. J. J. v. d. Burg and C. K. I. Williams, "An evaluation of change point detection algorithms," 2020. [Online]. Available: https://arxiv.org/abs/2003.06222

[15] "GitHub repository for diamond loss," https://github.com/caozhenxiang/multivie-TIRE.