REGULARIZATION VIA INVARIANT PATTERNS: TEMPORAL DOMAIN RANDOMIZATION FOR HUMAN ACTIVITY RECOGNITION

Anonymous authors
Paper under double-blind review

ABSTRACT

Synthetic data has become a common strategy to address data scarcity in Human Activity Recognition (HAR). However, models trained on synthetic samples often overfit to spurious features, leading to a substantial domain gap when transferred to real-world data. To address this challenge, we propose Regularization via Invariant Patterns (RIP), a novel data-centric method that extends the idea of domain randomization to the temporal domain. RIP augments time-series windows by "framing" them with invariant (constant-valued) patterns, compelling models to focus on informative signals rather than irrelevant temporal context. Evaluated across five HAR datasets, four classifiers, and more than 2,000 experiments, RIP consistently improves F1 scores, achieving gains of up to +53 percentage points (over +160% relative improvement) compared to synthetic baselines — often matching or surpassing real-data baselines. Beyond synthetic scenarios, RIP also boosts performance in real-only training settings, highlighting its broad applicability. Both theoretical analysis and empirical results show that RIP stabilizes weight updates and enhances calibration, all without modifying model architectures.

1 INTRODUCTION

Human Activity Recognition (HAR) increasingly uses synthetic samples to mitigate data scarcity, privacy constraints, and inter-subject variability, especially with wearable time-series data. However, models trained on synthetic data often break onto generator-specific cues, creating a sizeable synthetic-to-real gap at inference time Seib et al. (2020); Sankaranarayanan et al. (2018). Prior work indicates that the use of synthetic samples is most effective when combined with principled, data-centric regularization rather than used as a mere data augmentation strategy Souza et al. (2023); Lupión et al. (2024).

Background and Related Work. (1) *Domain randomization (DR).* In computer vision, it improves simulated-to-real transfer by randomizing non-essential factors (e.g., backgrounds) Tremblay et al. (2018). A principled *temporal* analogue for wearable HAR remains underexplored. (2) *Time-series/HAR regularization.* Common data-centric approaches include jitter, scaling, time-warping, permutation/cropping, and *temporal masking/cutout* (SpecAugment-style), as well as Mixup/Cutmix and optimization-level methods like SAM/DRO Zhang et al. (2018); Yun et al. (2019); Foret et al. (2021); Kuhn et al. (2024); Bento et al. (2023). These techniques perturb local dynamics or alter the loss, but generally do *not* enforce invariance to non-informative temporal context. (3) *Domain generalization (DG) and calibration.* DG methods operate on losses/representations rather than inputs and often require architectural or optimization changes Arjovsky et al. (2019).Most existing research has concentrated on specific modalities such as images and text, while applications to time-series data remain relatively underexplored Deng et al. (2024).

Our idea. We present *Regularization via Invariant Patterns (RIP)*, a simple, architecture-agnostic *temporal* DR mechanism: each training window is "framed" by *constant-valued* segments. RIP discourages reliance on spurious context and biases learning toward class-relevant dynamics by inducing class-agnostic invariance in the surrounding temporal context. Unlike zero/edge padding

or random masking, RIP uses *structured* invariant patterns drawn from a small set of γ values, explicitly operationalizing DR in time.

Contributions. (i) We introduce RIP, a data-centric regularizer that, to our knowledge, brings domain randomization to the temporal axis for HAR without modifying architectures or losses. (ii) Across five datasets, four classifiers, and more than 2,000 runs, RIP consistently improves S→R transfer (average macro-F1 gains ≈+53 pp up to ≈+81 pp) and also boosts TRTR, with tighter confidence intervals. (iii) We analyze why RIP stabilizes learning—reduced hidden-state variance and better probability calibration. (iv) We position RIP against time-series augmentations and DG baselines, and provide ablations on duplication factor i and constant design γ. Extended related work appears in Appendix A.

2 Regularization via Invariant Patterns (RIP)

Regularization via Invariant Patterns (RIP) introduces a new form of data-centric regularization inspired by the principle of domain randomization Tremblay et al. (2018). In their work, Tremblay et al. (2018) demonstrated that training on synthetic images with randomized backgrounds forces a model to become invariant to non-essential features, thus bridging the sim-to-real domain gap. We translate this core idea from the spatial domain of images to the temporal domain of HAR data. For a time-series window, we treat the surrounding temporal context as the "background." RIP implements this concept of temporal domain randomization by strategically "framing" the core signal with constant-valued windows. These invariant patterns, defined by a scalar γ, compel the model to focus on the dynamic, informative part of the signal, much like a picture frame draws attention to the image it contains. This process encourages learning more robust and generalizable representations from synthetic data.

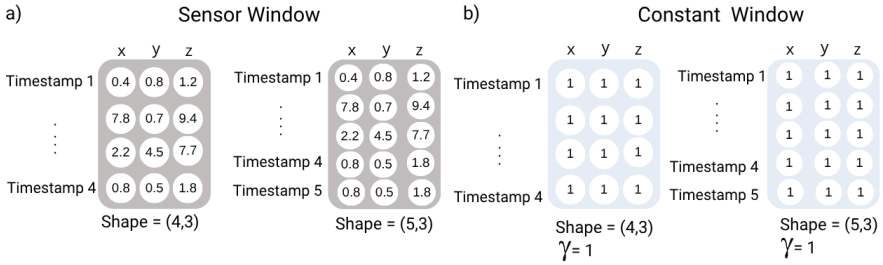

Figure 1: **Sensor vs. constant windows:** (a) Varying signals over time. (b) Fixed values across time and attributes are used for regularization. Each window has ω timestamps and 3 attributes.

Preliminary Concepts. Let $\mathcal{D}$ be a dataset of time-series samples, where each sample is a temporal window $x \in \mathbb{R}^{\omega \times 3}$. Here, $\omega \in \mathbb{N}$ is the window length, and the second dimension corresponds to the three sensor axes. We define two types of windows: a **sensor window**, containing dynamic sensor readings, and a **constant window**, where all values are fixed to a scalar $\gamma \in \mathbb{Z}$. This invariant pattern serves as the non-informative "temporal background" used to regularize the learning process. Figure 1 illustrates the structural difference between these two window types. The value of γ is a hyperparameter subject to tuning.

RIP Formalization. Given a dataset $\mathcal{D} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, the RIP method produces an augmented dataset $\mathcal{D}'$. For each sensor window x_i, we generate a new sample x_i' by creating a sequence of information where constant windows frame the original window. This operation is controlled by two hyperparameters: the constant value γ and a duplication factor $i \in \mathbb{N}$. The number of constant windows prepended and appended to form the sequence is defined as $2i$. The final augmented sample x_i' is a tensor of shape $(4i+1, \omega, 3)$, where ω is the original window length. The corresponding label y_i' is a sequence of the original label repeated $4i + 1$ times. The whole procedure is formally described in Algorithm 1.

By explicitly introducing invariant information into the training data, RIP compels the model to learn from the contrast between constant and dynamic temporal patterns. This strategy reinforces

Algorithm 1 Creating the RIP-Augmented Dataset D'

Input: Dataset $\mathcal{D} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, constant value γ, duplication factor $i \in \mathbb{N}$
Output: Augmented dataset $\mathcal{D}'$ where each sample is a sequence of windows.
Let $\mathcal{D}' \leftarrow \emptyset$
for each sample $(x_m, y_m) \in \mathcal{D}$ **do**
  Let $k \leftarrow 2i$ {Calculate the number of constant windows for each side}
  Let ω be the length of the window x_m
  Let $C \in \mathbb{R}^{\omega \times 3}$ be a constant window filled with the value γ
  Let $S_{\text{prefix}} \leftarrow \text{Repeat}(C, k)$ {Create a sequence of k constant windows}
  Let $S_{\text{suffix}} \leftarrow \text{Repeat}(C, k)$ {Create another sequence of k constant windows}
  Let $S \leftarrow S_{\text{prefix}} + [x_m] + S_{\text{suffix}}$ {Combine to form the final sequence of windows}
  $x'_m \leftarrow \text{Stack}(S)$ {Convert sequence to tensor of shape $(2k+1, \omega, 3)$}
  $y'_m \leftarrow \text{Repeat}(y_m, 2k+1)$ {Create corresponding label sequence}
  $\mathcal{D}' \leftarrow \mathcal{D}' \cup \{(x'_m, y'_m)\}$
end for
return $\mathcal{D}'$

Table 1: Performance comparison of RIP against the baseline under the Train on Synthetic, Test on Real (TSTR) protocol. We report the baseline for each dataset against the best-performing RIP configuration. The best result for each metric is shown in bold.

Model	**Dataset**	**RIP Config.**	**Accuracy (%)**		**F1-score (%)**	
		(γ, i)	Baseline	RIP	Baseline	RIP
DClassifier	MHEALTH	(0, 5)	55.79±2.44	**97.75±0.37**	52.56±2.33	**97.92±0.34**
	MHAD1	(0, 16)	33.86±1.46	**86.03±0.59**	32.85±1.66	**86.03±0.58**
	MHAD2	(0, 16)	46.97±3.70	**81.66±0.63**	43.52±3.78	**81.10±0.73**
	WHARF	(-1, 16)	15.46±3.04	**89.99±0.87**	6.14±1.78	**87.26±1.26**
	WISDM	(1, 5)	53.03±3.04	**99.79±0.05**	44.94±2.51	**99.66±0.08**
TS-Classifier	MHEALTH	(0, 1)	**61.00±4.09**	58.83±2.34	**57.42±4.38**	53.99±2.72
	MHAD1	(-1, 16)	**35.62±1.98**	29.92±1.02	**32.58±2.39**	25.99±1.12
	MHAD2	(-1, 16)	48.31±3.62	**51.44±2.85**	41.97±4.12	**41.50±3.32**
	WHARF	(-1, 1)	20.94±3.16	**44.90±4.10**	11.41±2.47	**26.61±2.07**
	WISDM	(1, 5)	50.07±3.78	**93.14±1.50**	47.38±2.49	**92.42±1.76**
TSBF	MHEALTH	(1, 1)	31.44±2.30	**31.50±2.38**	26.63±2.36	**26.79±2.10**
	MHAD1	(1, 5)	19.94±0.98	**20.32±0.71**	18.80±0.97	18.80±0.70
	MHAD2	(-1, 5)	37.46±2.03	**38.00±1.64**	33.19±2.27	**33.96±1.89**
	WHARF	(-1, 5)	**15.47±3.77**	15.31±3.61	5.53±1.76	5.43±1.52
	WISDM	(0, 16)	39.10±6.34	**39.39±6.54**	29.22±2.46	**30.58±2.88**
TSRF	MHEALTH	(-1, 5)	29.72±1.68	**29.91±2.42**	25.98±1.07	**26.52±1.75**
	MHAD1	(-1, 1)	21.51±0.84	**21.75±0.82**	19.59±0.97	**19.85±0.96**
	MHAD2	(-1, 5)	34.65±2.95	**36.15±2.43**	31.13±3.08	**32.77±2.65**
	WHARF	(0, 16)	12.60±3.39	**12.82±3.37**	4.05±1.09	**4.41±1.22**
	WISDM	(-1, 16)	29.58±6.75	**30.20±7.01**	24.14±4.21	**24.60±4.69**

the focus on meaningful signal patterns, reducing variance in the learned weights and enhancing the model's ability to generalize across both synthetic and real data distributions[1]

[1]The source code for the proposed method is publicly available at after_review_process.

Table 2: Performance comparison of RIP against the baseline on real-world data. The best result for each metric per dataset is shown in bold. The RIP results represent the best-performing hyperparameter configuration.

Model	Dataset	RIP Config.	Accuracy (%)		F1-score (%)	
		(γ, i)	Baseline	RIP	Baseline	RIP
DClassifier	MHEALTH	(0, 16)	91.21±1.61	**97.84±0.21**	90.46±2.84	**98.01±0.20**
	MHAD1	(0, 16)	58.25±2.06	**86.12±0.55**	67.13±1.14	**86.13±0.53**
	MHAD2	(0, 16)	68.32±1.05	**82.12±0.72**	67.58±1.31	**81.68±0.74**
	WHARF	(1, 16)	83.11±1.85	**90.25±1.19**	78.62±2.08	**87.48±1.70**
	WISDM	(0,5)	99.47±0.09	**99.77 ± 0.03**	99.20±0.15	**99.46 ± 0.05**
TS-Classifier	MHEALTH	(1,1)	32.37±2.40	**59.98±2.03**	24.08±2.59	**55.54±2.42**
	MHAD1	(-1,5)	20.45±1.15	**31.67±0.73**	15.67±1.41	**26.08±0.70**
	MHAD2	(1,16)	31.25±1.64	**52.49±3.16**	24.46±1.72	**43.22±3.91**
	WHARF	(1,1)	19.19±6.71	**46.06±4.34**	10.28±3.63	**28.63±2.24**
	WISDM	(-1,5)	90.32±1.63	**93.12±1.21**	87.19±2.62	**92.30±1.46**

3 Experimental Setup

To evaluate the effectiveness of our proposed RIP method, we conducted extensive experiments on both real and synthetically generated datasets. Our setup is designed to isolate the impact of RIP on synthetic data quality and assess its broader applicability.

Data and Models We used the Time-LogCosh-GAN (TLCGAN) Souza et al. (2023) to generate synthetic tri-axial accelerometer data for five publicly available HAR datasets: MHAD1 Chen et al. (2015), MHAD2 Chen et al. (2015), MHEALTH Banos et al. (2014), WISDM Weiss, and WHARF Bruno et al. (2013). For classification, we employed a diverse set of four models: Deep ConvLSTM (DClassifier) Singh et al. (2020), TS-Classifier hfawaz (2020), Time Series Random Forest (TSRF) for, and Time Series Bag-of-Features (TSBF) fea. Full details on datasets, preprocessing steps, and model implementations are provided in Appendix B.

Evaluation Protocol Our evaluation focuses on the Train on Synthetic, Test on Real (TSTR) protocol, which is particularly suited to assess whether synthetic data, when enhanced by RIP, effectively contributes to model generalization. As a baseline for real-world application, we also employed the conventional Train on Real, Test on Real (TRTR) protocol. Given the inherent class imbalance in HAR datasets, we report several metrics, primarily focusing on the F1 score due to its robustness. Performance improvements are consistently reported in percentage points over the respective baselines. For example, a baseline F1 score of 10% with an improvement of 4 percentage points results in a new score of 14% (details are provided in Appendix C).

Hyperparameters Our method introduces two hyperparameters: the constant value γ and the duplication factor i. In the synthetic data experiments, we evaluated $\gamma \in \{-1, 0, 1, 5\}$ and $i \in \{1, 5, 16\}$. The same configurations were tested on real datasets to assess RIP's general applicability. Values of γ within $[-1, 1]$ were chosen to match the data distribution, while $\gamma = 5$ was included as an out-of-range control to test robustness beyond the natural interval. A complete description of all hyperparameter settings and configurations is provided in Appendix B.

4 Results and Discussion

Results on TSTR. Table 1 summarizes the best-performing RIP configurations, demonstrating significant performance gains across multiple models and datasets. Our analysis, detailed in the Appendix C, reveals four key findings: (1) RIP disproportionately benefits deep learning models that rely on representation learning (DClassifier and TS-Classifier); (2) it enhances model fairness and robustness, not just predictive accuracy; (3) its effectiveness is highly dependent on the dataset characteristics; and (4) it can help bridge the synthetic-to-real performance gap.

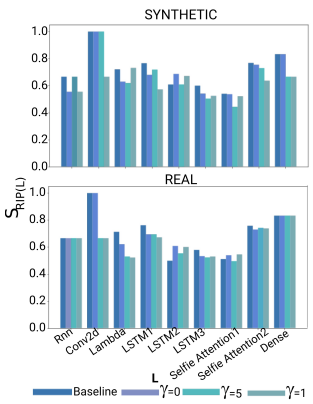

Figure 2: Layer-wise KS statistics for DClassifier weights trained on real and synthetic MHAD2 data, tested against a uniform distribution. Each color denotes a model; x-axis: layers, y-axis: KS value.

Overall findings. A clear pattern emerges across all experiments: RIP provides substantially stronger improvements for deep architectures (DClassifier and TS-Classifier) than their statistical counterparts (TSBF and TSRF). In some cases, RIP led to dramatic gains, such as over +53 percentage points in F1 (corresponding to more than $+160\%$ relative improvement) and even extreme cases of more than +1300% relative improvement. This strong correlation suggests that RIP's regularization mechanism is particularly beneficial for models engaged in representation learning, possibly by preventing overfitting to spurious features in the synthetic data.

Per-dataset variations. The datasets respond differently to RIP. For example, MHAD2 shows higher instability, with gains depending on the adopted configuration. In contrast, WHARF consistently benefits, reaching some of the most significant relative improvements observed. Regarding the hyperparameter i, the distribution of its optimal value (40% for i=16, 40% for i=5, and 20% for i=1) underscores that there is no single best setting. Instead, the required amount of regularization is a function of the dataset's complexity and the model's capacity, highlighting the necessity of treating i as a crucial hyperparameter to be tuned for each specific context.

Fairness and robustness. By consistently reducing the gap between Accuracy and macro F1-score, RIP mitigates bias against under-represented classes, leading to more balanced predictions. Furthermore, models trained with RIP exhibit significantly narrower confidence intervals across runs compared to the baseline (see Tab. 1), indicating increased training stability and more reliable performance.

Results on TRTR. Table 2 highlights the impact of RIP on real-world data, particularly for deep learning models. The results show consistent performance gains, alongside improvements in robustness and fairness.

Performance gains by model. RIP significantly improved both models, though in distinct ways. For DClassifier, which already had strong baselines, RIP consistently improved results to higher levels—for example, on MHAD1, Accuracy increased by +27.9 percentage points (pp) and F1 by +19.0 pp. On MHEALTH, F1 rose from 90.5% to 98.0%, setting a new performance bound. Despite weaker baselines, TS-Classifier achieved the most significant relative improvements: on MHEALTH, the F1 increased by +31.5 pp (a relative gain of 131%), and on WHARF by +18.4 pp (179%). These findings suggest that RIP is particularly effective at regularizing models that struggle to generalize.

Hyperparameter effects. Analysis of the optimal configurations shows that the duplication factor i is a dominant parameter: $i = 16$ was optimal in half of the cases, while $i = 5$ and $i = 1$ were also effective depending on dataset complexity. No single value of γ consistently prevailed, with $\gamma = 0$ and $\gamma = 1$ each appearing in 40% of the best cases, indicating sensitivity to the model–dataset interaction.

Robustness and fairness. Beyond accuracy, RIP narrowed the gap between Accuracy and F1, indicating better treatment of underrepresented classes. For instance, TS-Classifier on MHEALTH reduced the gap from 8 pp at baseline to 4 pp under RIP. Moreover, RIP consistently decreased standard deviations, yielding more stable and trustworthy performance. With DClassifier on MHEALTH, the Accuracy standard deviation dropped from ± 1.61 to ± 0.21, showing that RIP leads to more reliable training outcomes.

Additional observations. Tree-based models (TSRF and TSBF), reported in the Appendix, already achieved near-perfect performance ($> 99\%$ F1) under TSTR. RIP maintained or slightly improved these results, demonstrating that it does not degrade performance even in scenarios with little room for improvement. Overall, RIP emerges as a safe and effective regularizer that improves weaker models, stabilizes stronger ones, and contributes to fairer and more robust performance across datasets.

Computational Cost. Applying RIP is primarily influenced by the dataset size and the duplication factor i. As an illustrative example, applying RIP with $i = 16$ to the WISDM dataset on a CPU-based model increased runtime from 2 hours to 4 hours and memory usage from 4 GB to 10 GB. In contrast, using a GPU for DL models reduced the training time for the same configuration to approximately 30 minutes, with a comparable memory usage. More details in Appendix C.

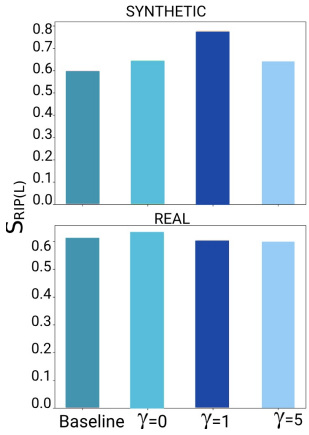

Figure 3: KS statistics between model output logits (real vs. synthetic MHAD2) and a uniform distribution. Lower values indicate outputs closer to uniform. Each bar shows a different model.

Regularization Effect. To analyze RIP's effect as a regularizer, we conducted a layer-wise study of DClassifier weight distributions, measuring their divergence from a uniform reference using Kolmogorov–Smirnov (KS) and Wasserstein distances (see Appendix D for setup). Results (Figs. 2, 3) show that RIP, particularly with $\gamma = 1$, consistently drives weights toward greater uniformity, especially in contextual layers such as SelfAttention and LSTM. This mechanism is analogous in purpose but distinct in effect from traditional ℓ_1 and ℓ_2 regularization. While ℓ_1 and ℓ_2 penalize large weights to induce sparsity (ℓ_1) or a narrow, zero-centered distribution (ℓ_2), RIP promotes a uniform distribution. This encourages the model to evenly utilize a wide range of weight values rather than concentrating them around zero. This uniformity is associated with reduced overfitting and improved calibration of output confidence. In contrast, $\gamma = 5$ produces stronger but less stable perturbations, while $\gamma = 0$ shows a reasonable regularization effect. These findings reveal a key trade-off: RIP improves local weight uniformity but may distort global structure if γ is set too high. Overall, $\gamma = 1$ offers the most robust balance, yielding models that are more decisive and reliable across datasets. A more detailed analysis of layer-wise dynamics is presented in Appendix D.

Table 3: Performance analysis of the proposed RIP method versus a baseline approach and the ℓ_1 and ℓ_2 regularization methods. Experiments were conducted on five public, real-world datasets. For each metric, the highest-performing result is highlighted in bold. The reported performance for RIP is based on its best-found hyperparameter configuration.

Model	Dataset	Method	Accuracy (%)	F1-score (%)
TS-Classifier	MHEALTH	Baseline	32.37±2.40	24.08±2.59
		RIP ($\gamma = 1, i = 1$)	**59.98±2.03**	**55.54±2.42**
		ℓ_1 ($\epsilon = 1.0$)	57.71±2.60	51.67±3.56
		ℓ_2 ($\epsilon = 0.1$)	58.25±3.37	51.61±4.45
	MHAD1	Baseline	20.45±1.15	15.67±1.41
		RIP ($\gamma = -1, i = 5$)	31.67±0.73	26.08±0.70
		ℓ_1 ($\epsilon = 0.001$)	31.49±1.95	27.62±2.00
		ℓ_2 ($\epsilon = 0.001$)	**33.42±1.95**	**29.70±2.29**
	MHAD2	Baseline	31.25±1.64	24.46±1.72
		RIP ($\gamma = 1, i = 16$)	**52.49±3.16**	**43.22±3.91**
		ℓ_1 ($\epsilon = 0.001$)	47.13±6.13	40.94±6.24
		ℓ_2 ($\epsilon = 0.001$)	44.65±6.04	37.84±6.03
	WHARF	Baseline	19.19±6.71	10.28±3.63
		RIP ($\gamma = 1, i = 1$)	**46.06±4.34**	**28.63±2.24**
		ℓ_1 ($\epsilon = 0.001$)	19.51±2.09	10.92±1.98
		ℓ_2 ($\epsilon = 0.001$)	20.04±3.16	10.90±2.73
	WISDM	Baseline	90.32±1.63	87.19±2.62
		RIP ($\gamma = -1, i = 5$)	**93.12±1.21**	**92.30±1.46**
		ℓ_1 ($\epsilon = 0.001$)	43.68±5.46	43.94±4.06
		ℓ_2 ($\epsilon = 0.001$)	47.88±6.81	46.12±3.21

5 ADDITIONAL EXPERIMENTS

RIP vs. Traditional Regularization. To assess the value of RIP as a general-purpose regularization method for HAR, we compared it against standard ℓ_1 and ℓ_2 techniques across both real and synthetic datasets, using two architectures (TS-Classifier and DClassifier). Table 3 consistently showed that RIP either matched or outperformed traditional methods, particularly in challenging or high-baseline scenarios. RIP was more robust across datasets with varying complexity, preserving or improving performance even where ℓ_1 and ℓ_2 caused degradation—up to 50 percentage points in some cases. It also proved more versatile, delivering consistent gains across a wide range of γ and i configurations, without requiring extensive tuning. RIP demonstrated generalization across architectures, showing both models' effectiveness, while traditional regularizers offered only marginal or inconsistent improvements. RIP provides a stable and straightforward alternative in scenarios where regularization is needed but domain-specific tuning is impractical. Due to space limitations, we have exclusively presented the results for the TS-Classifier within the main body of this paper, as they most accurately reflect the overall behavior observed in our experiments. Comprehensive results, methodological justifications, and all corresponding tables are available in Appendix F.

RIP vs. HAR approaches. Our literature review revealed a scarcity of research addressing regularization and domain generalization specifically for HAR. The most relevant prior work, Bento et al. (2023), explored Mixup and Distributionally Robust Optimization (DRO) for accelerometer-based HAR, partially aligning with our objectives. To ensure a fair comparison, we benchmarked RIP against these approaches, along with Cutmix Yun et al. (2019), DRO and Mixup Zhang et al. (2018) using the MHAD2 dataset. For brevity, only the best-performing configuration of each competing method is reported in Tab. 4. A complete set of results, parameter sweeps, and detailed commentary is provided in the Appendix F.

Superior Performance and Reliability. Across all metrics, RIP consistently outperforms the baseline and competing regularizers. Beyond delivering higher Accuracy and F1-scores, RIP exhibits enhanced reliability, as evidenced by its significantly narrower confidence intervals (see Table 4). This reduced variance indicates more stable training dynamics and more trustworthy predictions, highlighting RIP as more accurate and dependable.

Fairness and the Synthetic-to-Real Gap Competing methods on synthetic data often display a wide disparity between Accuracy and F1, indicating bias against minority classes. RIP maintains a close alignment between these metrics, yielding more equitable predictions. This disparity is less pronounced on real datasets, but RIP remains competitive with the best-performing methods, reinforcing its ability to promote fairer outcomes. While methods like Cutmix achieve moderate performance on synthetic data, they degrade substantially when transferred to real data, exacerbating the domain gap. In contrast, RIP consistently delivers state-of-the-art results in both domains, demonstrating unique strength in bridging this critical divide.

RIP in Other Domains. We conducted preliminary experiments on general time-series and tabular datasets to explore RIP's applicability beyond wearable sensor data. Results suggest that RIP's most substantial benefits arise in structured, repetitive sensor settings, while improvements in other domains are modest or inconsistent. Significantly, performance rarely degrades substantially, indicating that RIP is a safe-to-try regularizer even outside its primary target. These findings reinforce RIP's specialization for HAR while also pointing toward future research opportunities, such as adapting the invariant framing principle to multimodal or irregular time-series domains. For completeness, we report and discuss detailed per-domain results in the Appendix F. Due to space constraints, here we present only a summarized overview.

6 THEORETICAL ANALYSIS

Previous analyses suggested that context-based architectures are more susceptible to the effects of RIP. This observation is particularly evident in our experiments for classifiers with recurrent designs. To provide a theoretical perspective, we therefore consider the case of a Recurrent Neural Network (RNN) with bias $b \in \mathbb{R}^h$ and identity activation $\varphi(x) = x$. At time step t, the input is $x_t \in \mathbb{R}^d$ with target $y_t \in \mathbb{R}^q$. The hidden state and output are given by $h_t = Wx_t + Uh_{t-1} + b$, $o_t = Vh_t + C$, $\hat{y}_t = \text{softmax}(o_t)$, where $W \in \mathbb{R}^{d\times h}$, $U \in \mathbb{R}^{h\times h}$, $V \in \mathbb{R}^{h\times q}$ and $C \in \mathbb{R}^q$. We simulate a sequence of 5 steps, with $h_0 = 0$, where constant samples γ are injected at $t = 1, 2, 4, 5$ (see Appendix E).

As derived in the Appendix, the hidden state at $t = 5$ under RIP is

$$h_5^{(\text{RIP})} = (U^4 + U^3 + U + I)W\gamma + U^2 W x_3 + \sum_{i=0}^{4} U^i b, \tag{1}$$

where $x_1 = x_2 = x_4 = x_5 = \gamma$. In contrast, without RIP, we have

$$h_5^{(\text{noRIP})} = \sum_{i=0}^{4} \left(U^i W x_{5-i} + U^i b \right). \tag{2}$$

Equations 4–2 show that RIP constrains h_5, since most dynamics arise from powers of U acting on the fixed term $W\gamma$. Assuming i.i.d. inputs with mean μ and covariance Σ, the expectations and variances are

$$\mathbb{E}[h_5^{(\text{noRIP})}] = \sum_{i=0}^{4} U^i W\mu + \sum_{i=0}^{4} U^i b, \quad \text{Var}(h_5^{(\text{noRIP})}) = \sum_{i=0}^{4} U^i W\Sigma W^\top (U^i)^\top,$$

$$\mathbb{E}[h_5^{(\text{RIP})}] = (U^4 + U^3 + U + I)W\gamma + U^2 W\mu + \sum_{i=0}^{4} U^i b, \quad \text{Var}(h_5^{(\text{RIP})}) = U^2 W\Sigma W^\top (U^2)^\top.$$

Thus, while $\text{Var}(h_5^{(\text{noRIP})})$ aggregates variability from five independent sources, $\text{Var}(h_5^{(\text{RIP})})$ depends only on x_3. RIP therefore reduces temporal diversity and constrains the hidden representation to a lower-variance subspace. This implicit regularization yields smoother gradients and more stable weight updates: $\Omega \leftarrow \Omega - \alpha \frac{\partial \ell}{\partial \Omega}, \qquad \Omega \in \{W, U, V, b\}$, where $\frac{\partial \ell}{\partial \Omega}$ inherits the reduced variability of h_5. While this promotes generalization, excessive repetition (large duplication factor i) can over-constrain the model and limit representational capacity.

Table 4: Performance comparison demonstrating the superiority of our method over the baseline and other standard techniques. The best results per protocol are shown in bold.

Method	**Accuracy (%)**	**F1-Score (%)**	**Epsilon**	**Protocol**
Baseline	46.97±3.70	43.52±3.78	-	
Cutmix	45.38±2.83	41.80±4.05	0.1	TSTR
DRO	32.70±4.71	26.88±4.99	0.3	
Mixup	39.88±6.14	32.33±9.63	0.1	
RIP (γ=0, i=16)	**81.66 ±0.63**	**81.10 ±0.73**	-	
Baseline	68.32±1.05	67.58±1.31	-	
Cutmix	63.10±1.18	62.03±1.47	0.1	TRTR
DRO	45.75±2.08	38.91±2.76	0.3	
Mixup	39.98±12.17	30.89±16.12	0.1	
RIP (γ =0, i=16)	**82.12 ± 0.72**	**81.68 ± 0.74**	-	

7 ABLATIONS

To better understand the core mechanisms behind our proposed RIP method, we conducted a series of ablation studies addressing three key questions. While we report high-level findings here, complete experimental setups and extended results are detailed in Appendix G.

Can naive data duplication achieve similar effects to invariant patterns? Not entirely. Duplicating input windows (e.g., $i = 5$) provides marginal improvements over the TSTR baseline, but these gains are not statistically significant and vanish for larger i. This indicates that naive repetition may induce overfitting, limiting generalization—especially for time-series, where subtle variations are critical.

Is i as a duplication factor necessary? Yes. When using a minimal RIP structure with only one central window (i.e., $\tilde{i} = \frac{1}{2}i$), performance drops below the synthetic baseline. Full duplication

patterns (e.g., $i = 5$) result in substantial gains—up to 4 percentage points in F1 score—confirming that structural repetition enhances the contextual framing effect and model focus.

Can random distributions replace constants? No. Replacing γ with non-stationary random values (e.g., rand$(0, 1)$ or rand$()$) consistently degrades performance. Even the best randomized setup merely matches the baseline. These results suggest that randomness introduces spurious patterns, confusing the model rather than improving robustness.

Does structure and γ-design matter? Absolutely. Experiments with unordered or non-integer γ (e.g., $\gamma = \text{Avg(features)}$) led to performance degradation. This confirms that the contextual frame's order and fixed design are essential for RIP's effectiveness. These findings highlight that the benefits of RIP arise not from trivial data augmentation or randomness, but from the careful design of its structure and components. Additional experiment details, tables, and visualizations are provided in Appendix G.

8 Conclusion

We introduced RIP, a data-centric regularization strategy for tri-axial wearable sensor data. By augmenting training datasets with invariant patterns, RIP improves generalization without modifying model architectures or loss functions. Our extensive experiments show that RIP enhances the performance and calibration of deep learning models, particularly in human activity recognition (HAR) tasks, offering both synthetic and real data improvements. RIP effectively addresses challenges in wearable data, such as scarcity, variability, and noise. It reduces weight variance and overconfidence, leading to more uniform weight distributions and better-calibrated predictions—especially in deep models. Mathematically, RIP modifies the optimization landscape by shifting update dynamics at the batch level. We demonstrated that its effects cannot be reproduced by naively holding most batch samples constant. The whole structure of RIP is necessary for its regularization effect. Among its hyperparameters, the duplication factor i proved most influential, with $i = 16$ yielding consistently strong results. While RIP shows limited effectiveness outside its target domain—failing to generalize to generic time-series or tabular data—it remains a lightweight, architecture-agnostic technique with high practical value. It requires no architectural changes, making it accessible to practitioners seeking to improve model robustness on sensor-based datasets. RIP extends the domain randomization applicability and holds promise for domains where reliable human activity data is essential, such as healthcare monitoring, sports analytics, and eldercare systems. Its simplicity, empirical effectiveness, and interpretability make it a valuable addition to the HAR modeling toolbox. Future work may explore its role in multimodal sensor setups, where data complexity further amplifies the need for effective regularization.

Reproducibility Statement

The source code associated with this work will be released on GitHub upon completion of the review process. Our implementation is developed on top of the TensorFlow framework, and we explicitly reference any external code utilized, including the classifiers incorporated, which were not reimplemented from scratch. A comprehensive description of all hyperparameters is provided in the Appendix, and the corresponding configuration files are included in the source code repository to facilitate reproducibility.

References

Time series bag-of-features. URL https://pyts.readthedocs.io/en/latest/auto_examples/classification/plot_tsbf.html#sphx-glr-auto-examples-classification-plot-tsbf-py. Accessed: 2023-04-04.

Time series forest. URL https://pyts.readthedocs.io/en/latest/generated/pyts.classification.TimeSeriesForest.html#pyts.classification.TimeSeriesForest. Accessed: 2023-04-04.

TensorFlow API Docs: tf.keras.layers.LSTM. URL https://www.tensorflow.org/api_docs/python/tf/keras/layers/LSTM. Accessed on: 11 Agust 2023.

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv*, 2019.

Oresti Banos, Rafael Garcia, and Alejandro Saez. MHEALTH Dataset, 2014. DOI: https://doi.org/10.24432/C5TW22.

Nuno Bento, Joana Rebelo, Andre V Carreiro, Francois Ravache, and Marilia Barandas. Exploring regularization methods for domain generalization in accelerometer-based human activity recognition. *Sensors*, 23(14):6511, 2023.

S.D. Brown and A.J. Myles. 3.17 - decision tree modeling in classification. In Steven D. Brown, Romá Tauler, and Beata Walczak (eds.), *Comprehensive Chemometrics*, pp. 541–569. Elsevier, Oxford, 2009. ISBN 978-0-444-52701-1. doi: 10.1016/B978-044452701-1.00025-9.

Barbara Bruno, Fulvio Mastrogiovanni, Antonio Sgorbissa, Tullio Vernazza, and Renato Zaccaria. Analysis of human behavior recognition algorithms based on acceleration data. *2013 IEEE International Conference on Robotics and Automation*, pp. 1602–1607, 2013. URL https://api.semanticscholar.org/CorpusID:18022557.

Luis Candanedo. Occupancy Detection . UCI Machine Learning Repository, 2016. DOI: https://doi.org/10.24432/C5X01N.

Chen Chen, Roozbeh Jafari, and Nasser Kehtarnavaz. Utd-mhad: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor. In *2015 ICIP*, pp. 168–172, 2015. doi: 10.1109/ICIP.2015.7350781.

Zicun Cong, Lingyang Chu, Yu Yang, and Jian Pei. Comprehensible counterfactual explanation on kolmogorov-smirnov test, 2021.

Paulo Cortez, A. Cerdeira, F. Almeida, T Matos, and J. Reis. Wine Quality, 2009. DOI: https://doi.org/10.24432/C56S3T.

Songgaojun Deng, Olivier Sprangers, Ming Li, Sebastian Schelter, and Maarten de Rijke. Domain generalization in time series forecasting. *ACM Trans. Knowl. Discov. Data*, 18(5), February 2024. ISSN 1556-4681. doi: 10.1145/3643035. URL https://doi.org/10.1145/3643035.

Mohammad Fekri, Ananda Mohon Ghosh, and Katarina Grolinger. Generating energy data for machine learning with recurrent generative adversarial networks. *Energies*, 13, 12 2019. doi: 10.3390/en13010130.

R. A. Fisher. Iris, 1988. DOI: https://doi.org/10.24432/C56C76.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=6Tm1mposlrM.

B. German. Glass Identification, 1987. DOI: https://doi.org/10.24432/C5WW2P.

Kevin J. Grimm, Gina L. Mazza, and Pega Davoudzadeh. Model selection in finite mixture models: A k-fold cross-validation approach. *Structural Equation Modeling: A Multidisciplinary Journal*, 24(2):246–256, 2017. doi: 10.1080/10705511.2016.1250638. URL https://doi.org/10.1080/10705511.2016.1250638.

Jeff Heaton. Ian goodfellow, yoshua bengio, and aaron courville: Deep learning. *Genetic Programming and Evolvable Machines*, 19(1):305–307, 2018. ISSN 1573-7632. doi: 10.1007/s10710-017-9314-z. URL https://doi.org/10.1007/s10710-017-9314-z.

Nadine Helwig, Enrico Pignanelli, and Andreas Schtze. Condition monitoring of hydraulic systems. Dataset, 2015. URL https://doi.org/10.24432/C5CW21.

hfawaz. Time-series classification from scratch, 2020. URL https://github.com/keras-team/keras-io/tree/master/examples/timeseries.

A. E. Hoerl and R. W. Kennard. Ridge regression: Application to nonorthogonal problems. *Technometrics*, 12(1):69–82, 1970. doi: 10.1080/00401706.1970.10488635. URL http://dx.doi.org/10.1080/00401706.1970.10488635.

Wenxin Jiang, Vishnu Banna, Naveen Vivek, Abhinav Goel, Nicholas Synovic, George K Thiruvathukal, and James C Davis. Challenges and practices of deep learning model reengineering: A case study on computer vision. *Empirical Software Engineering*, 29(6):142, 2024.

Michael Kahn. Diabetes. DOI: https://doi.org/10.24432/C5T59G.

Daniel Kuhn, Soroosh Shafiee, and Wolfram Wiesemann. Distributionally robust optimization, 2024. URL https://arxiv.org/abs/2411.02549.

Qiang Liu, Jiade Zhang, Jingna Liu, and Zhi Yang. Feature extraction and classification algorithm, which one is more essential? an experimental study on a specific task of vibration signal diagnosis. *To be defined if missing; e.g., International Journal of Signal Processing*, 2022.

Marcos Lupión, Federico Cruciani, Ian Cleland, Chris Nugent, and Pilar M. Ortigosa. Data augmentation for human activity recognition with generative adversarial networks. *IEEE Journal of Biomedical and Health Informatics*, 28(4):2350–2361, 2024. doi: 10.1109/JBHI.2024.3364910.

Andrea Martiniano and Ricardo Ferreira. Absenteeism at work. UCI Machine Learning Repository, 2018. DOI: https://doi.org/10.24432/C5X882.

S. Moro, P. Rita, and P. Cortez. Bank Marketing, 2012. DOI: https://doi.org/10.24432/C5K306.

Victor M. Panaretos and Yoav Zemel. Statistical aspects of wasserstein distances. *Annual Review of Statistics and Its Application*, 6(1):405–431, 3 2019. doi: 10.1146/annurev-statistics-030718-104938. URL https://doi.org/10.1146%2Fannurev-statistics-030718-104938.

Oliver Roesler. EEG Eye State. UCI Machine Learning Repository, 2013. DOI: https://doi.org/10.24432/C57G7J.

Ryan A. Rossi and Nesreen K. Ahmed. The network data repository with interactive graph analytics and visualization. In *AAAI*, 2015. URL https://networkrepository.com.

Swami Sankaranarayanan, Yogesh Balaji, Arpit Jain, Ser Nam Lim, and Rama Chellappa. Learning from synthetic data: Addressing domain shift for semantic segmentation. In *2018 IEEE/CVF*, pp. 3752–3761, 2018. doi: 10.1109/CVPR.2018.00395.

Viktor Seib, Benjamin Lange, and Stefan Wirtz. Mixing real and synthetic data to enhance neural network training–a review of current approaches. *arXiv:2007.08781*, 2020.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. Ai models collapse when trained on recursively generated data. *Nature*, 631(8022):755–759, 2024.

Satya P Singh, Madan Kumar Sharma, Aimé Lay-Ekuakille, Deepak Gangwar, and Sukrit Gupta. Deep convlstm with self-attention for human activity decoding using wearable sensors. *IEEE Sensors Journal*, 21(6):8575–8582, 2020.

M. D. Souza, C. R. S. Junior, J. Quintino, F. Q. B. Silva A.L. Santos, and C. Zanchettin. Exploring the impact of synthetic data on human activity recognition tasks. *Procedia Computer Science*, 2023. (in press).

Yingjie Tian and Yuqi Zhang. A comprehensive survey on regularization strategies in machine learning. *Information Fusion*, 80:146–166, 2022. ISSN 1566-2535. doi: https://doi.org/10.1016/j.inffus.2021.11.005. URL https://www.sciencedirect.com/science/article/pii/S156625352100230X.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996. ISSN 00359246. URL http://www.jstor.org/stable/2346178.

Jonathan Tremblay, Aayush Prakash, David Acuna, Mark Brophy, Varun Jampani, Cem Anil, Thang To, Eric Cameracci, Shaad Boochoon, and Stan Birchfield. Training deep networks with synthetic data: Bridging the reality gap by domain randomization. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 969–977, 2018.

Gary M. Weiss. *WISDM Smartphone and Smartwatch Activity and Biometrics Dataset*. Department of Computer and Information Science, Fordham University.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6023–6032, 2019.

Guodong Zhang, Chaoqi Wang, Bowen Xu, and Roger Grosse. Three mechanisms of weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=B1lz-3Rct7.

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=r1Ddp1-Rb.