

Article

Frequency–Time Domain Alignment for Cross-Subject Time-Series Classification Under Distribution Shift

Dong-Hyun Won ¹ and Kwang-Seong Shin ^{2,*}¹ Office of Liberal Arts Education, Wonkwang University, Iksan 54538, Republic of Korea; dhwon79@wku.ac.kr² Department of Computer Engineering, Sunchon National University, Suncheon 57992, Republic of Korea

* Correspondence: waver@scnu.ac.kr

Abstract

Background: Distribution shift is one of the most pervasive data-related challenges for time-series classification. In sensor-based Human Activity Recognition (HAR), a model trained on one population of subjects often degrades substantially when deployed on another, even with identical hardware and activity labels. This paper addresses that setting specifically: cross-subject distribution shift. **Methods:** We propose FTDA (Frequency–Time Domain Alignment), an unsupervised domain-adaptation method that processes raw signals and their FFT-magnitude spectra through a dual-branch encoder, combines a Gradient-Reversal adversarial loss on the joint feature with a Multi-Kernel MMD on the frequency branch, and applies a symmetric-KL time–frequency consistency loss on unlabeled target data. **Results:** On UCI HAR, over four cross-subject transfer tasks and 10 random seeds ($n = 40$ paired observations), FTDA reaches 95.01% target accuracy against 92.46% for Source-only, 92.29% for DANN and 93.32% for a RAINCOAT-style baseline. Because the tasks share subjects and two of them are mirror configurations, no pooled significance test is offered; inference is confined to within-task paired tests (10 seeds each, Holm-corrected), under which FTDA is ahead of every single-view baseline on all four tasks, while the difference from the two dual-view competitors (+1.69 pp against RAINCOAT-style, descriptive task-level interval $[-0.40, +3.78]$) is not established. Over a common set of seeds, controls widened to FTDA's parameter and multiply-accumulate budgets shifted accuracy by at most +0.35 pp against DANN (intervals including zero), where FTDA gains +2.97 pp on the same runs, so added capacity alone does not appear to explain the improvement, and a target-only reference model, which is optimistic because it is evaluated on the windows it was trained on, sits 0.70 pp above FTDA. An ablation shows that the frequency branch is useful chiefly because it enables a coupled mechanism of cross-view consistency and auxiliary source supervision (+2.02 pp for that final step) rather than concatenated features (+1.53 pp); for the frequency MMD term, no measurable benefit was detected (-0.57 pp, 95% CI $[-1.28, +0.13]$), and it can be removed. On a second dataset, HHAR, the cross-subject advantage does not reproduce under cross-device shift: we detect no difference from time-only adversarial baselines (paired difference +0.48 pp against DANN, 95% CI $[-1.20, +2.16]$; exploratory, since all device pairs share the same subjects), so the benefit of the spectral view does not appear to carry across shift types. **Conclusions:** Under cross-subject shift, the operative ingredient is not the spectral view as an extra set of features but the coupled consistency-and-auxiliary mechanism that the second view makes possible; the alignment objectives themselves contribute little, and no measurable benefit was detected for the frequency MMD term. FTDA realizes this view at a modest 0.31 M/7.55 MMAC footprint with no OT solver in the training loop and no hyper-parameter that requires target labels. Transfer across devices and operating environments is not evaluated here and remains open for future work.



Academic Editor: Jichai Jeong

Received: 17 June 2026

Revised: 4 August 2026

Accepted: 5 August 2026

Published: 6 August 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: domain adaptation; cross-subject distribution shift; time series; human activity recognition; frequency-domain learning; sensor data

1. Introduction

Time-series classifiers deployed on wearable and embedded sensors routinely encounter inputs whose statistical properties drift away from the conditions of training. Such drift arises across subjects (different body shapes, gait, and dexterity), across devices (sampling-rate variations and axis orientation), and across environments (temperature and mounting position), each changing the marginal distribution $p(x)$, and frequently, the conditional $p(y | x)$. This covariate shift is the canonical data-related challenge for machine learning in sensing and has been the subject of an active research line under the umbrella of unsupervised domain adaptation (UDA) [1,2]. Of these, the present study evaluates the cross-subject case only: all experiments hold the device and the recording protocol fixed and vary the population of subjects. Claims in this paper should be read as claims about cross-subject distribution shift; device-level and environment-level transfer are discussed as motivations and are revisited in Section 5 as limitations.

Most UDA methods for time series adapt techniques originally designed for vision: domain-adversarial networks (DANNs) [1], Maximum Mean Discrepancy (MMD) feature alignment [3], or self-supervised pre-training. In the time-series setting, however, periodic and rhythmic structure carries a large fraction of the discriminative signal—the principal cadence of walking, the spectral profile of an ECG complex, or the bandwidth of vibration in an industrial machine. When the gait of a target subject differs from the source, we expect that difference to appear in the frequency domain—in cadence and in the shape of the spectrum—rather than only in raw waveform amplitude. This is the hypothesis on which the method is built on; Section 4.12 reports a setting in which it does not hold. Two recent works—RAINCOAT [4] and TF-C [5]—confirm that the frequency view of time series contains adaptation-relevant information that time-domain features alone fail to exploit.

This paper develops a simple and reproducible method that explicitly aligns source and target distributions in both the time domain and the frequency domain and enforces semantic agreement between the two views on unlabeled target data. Our specific contributions are the following:

1. We introduce FTDA, a dual-branch encoder that processes raw signals and FFT-magnitude spectra in parallel and combines a domain-adversarial loss on the joint feature with a Multi-Kernel MMD loss on the frequency feature.
2. We add an unlabeled-target cross-view consistency loss between time-only and frequency-only classifiers, derived from the symmetric KL divergence between their softmax outputs.
3. We evaluate FTDA on the UCI HAR benchmark [6] under four cross-subject transfer tasks against Source-only, DANN, CoDATS-style [7], a capacity-matched time-only DANN control that isolates the frequency view from added model capacity, and three frequency-aware competitors: a frequency-only DANN, a RAINCOAT-style baseline [4] whose alignment term is an entropic-OT divergence, and a TF-C-style baseline [5] using time–frequency contrastive pre-training.
4. We provide a four-step component ablation, an ablation of the spectral preprocessing (log compression, spectral normalization, and DC handling), an ablation establishing that the auxiliary supervision term is what keeps the consistency loss non-vacuous, a hyper-parameter-sensitivity sweep, and statistical-significance tests on every method vs. FTDA comparison.

5. We release the full codebase, configuration JSONs, every per-run JSON result (96 runs), feature embeddings, and a clickable DOI-annotated reference list ([paper/references.html](https://doi.org/10.3390/electronics15153488)) so that every claim can be independently verified.

2. Related Work

We organize prior work along the four mechanisms that are actually available for closing a domain gap in time series, since methods that look superficially different often differ only in which of these they instantiate.

(i) Adversarial alignment.

DANN [1] introduced the Gradient Reversal Layer (GRL) and a domain discriminator that minimizes a proxy of the $\mathcal{H}$ -divergence that Ben-David et al. [2] showed bounds target risk. ADDA [8] untied the two encoders and replaced the GRL with a GAN-style objective, and CDAN [9] conditioned the discriminator on the classifier output so that alignment respects class structure. In time series, VRADA [10] couples this idea with a variational RNN, and CoDATS [7] combines convolutional features with adversarial alignment for HAR.

(ii) Discrepancy-based alignment.

Rather than train a discriminator, this family minimizes a closed-form distance between feature distributions: MMD in DAN [3] and in its kernel two-sample formulation [11], second-order statistics in Deep CORAL [12], and Wasserstein distance in WDGR [13]. Optimal transport gives the same family a geometric reading [14], made tractable by entropic regularization [15]; RAINCOAT [4] builds on this line for time series. AdvSKM [16] is directly relevant to our setting: it argues that ordinary MMD captures only low-order local statistics and is, therefore, ill-suited to non-stationary time series and reforms the MMD kernel spectrally in response.

(iii) Self-supervised time–frequency representation learning.

A third family sidesteps explicit alignment and instead learns transferable representations before adapting. TS2Vec [17] and TS-TCC [18] learn contrastive representations from unlabeled series, and TF-C [5] makes the time and frequency views of the same window a positive pair so that time–frequency agreement is used as a pre-training signal. CLUDA [19] combines contrastive representation learning with adversarial alignment within a single UDA framework.

(iv) Consistency regularization.

Finally, one may constrain the model’s own predictions on unlabeled data to be stable under perturbation or across views, as in Mean Teacher [20]; TENT [21] pushes this to inference time by minimizing prediction entropy on the test stream.

Positioning of this work.

FTDA belongs to (i), (ii), and (iv) simultaneously, and Table 1 makes the comparison concrete. Two distinctions matter. First, TF-C uses time–frequency agreement as a self-supervised pre-training objective on individual domains, whereas we use it as a discriminative consistency constraint on unlabeled target predictions during adaptation; the quantity being made consistent is a class posterior, not an embedding. Second, RAINCOAT addresses the harder universal-DA problem in which the label sets may differ, and it devotes much of its machinery to detecting and correcting private target classes; in the closed-set setting studied here, that machinery is inapplicable, and what remains is a dual time–frequency encoder aligned by optimal transport. Our contribution is, therefore, not

the discovery that spectral information helps—[4,5,16] establish that—but a controlled account of which ingredient does the work, together with a formulation that reaches the same accuracy without an OT solver in the training loop. Section 4.10 evaluates RAINCOAT-style and TF-C-style baselines directly rather than resting on this argument.

Table 1. Positioning of FTDA against the most closely related methods. “Consistency” distinguishes agreement enforced on embeddings during pre-training from agreement enforced on class posteriors of unlabeled target data during adaptation.

Method	Spectral View	Alignment	Consistency	Setting
DANN [1]	—	adversarial	—	closed-set UDA
DAN [3]	—	MMD	—	closed-set UDA
CoDATS [7]	—	adversarial	—	multi-source HAR
AdvSKM [16]	kernel	spectral MMD	—	closed-set UDA
TF-C [5]	yes	—	embedding (pre-train)	self-supervised transfer
RAINCOAT [4]	yes	optimal transport	—	universal DA
CLUDA [19]	—	adversarial	embedding	closed-set UDA
FTDA (ours)	yes	adversarial + MMD	class posterior (target)	closed-set UDA

3. Materials and Methods

3.1. Problem Setting

Let $\mathcal{D}_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ denote labeled source data and $\mathcal{D}_t = \{x_j^t\}_{j=1}^{n_t}$ denote unlabeled target data, with $x \in \mathbb{R}^{C \times L}$ being a multi-channel window of length L with C channels. Labels $y \in \{1, \dots, K\}$ are shared across domains. We seek a hypothesis minimizing target risk.

3.2. Dual-Branch Encoder

The time branch g_t is a 1-D CNN with three convolutional blocks (channels [64, 128, 128], kernels [8, 5, 3], strides [2, 2, 2], batch-norm, ReLU, and dropout), adaptive average pooling, and a linear projection to $\mathbb{R}^d$ with $d = 128$. The frequency branch g_f has the same depth but operates on the magnitude spectrum with stride-1 convolutions. The joint feature $z = h([g_t(x); g_f(x)])$ is produced by a small fusion MLP h . Figure 1 shows the resulting architecture and the attachment point of every loss term.

Spectral preprocessing.

Because the reported benefit rests on the frequency view, we state the transform exactly. Windows are first z-normalized per channel over the source training set (the same statistics are then applied to the target, computed without target labels). The frequency branch consumes

$$S(x) = |\text{rFFT}(x)| \in \mathbb{R}^{C \times (L/2+1)}, \quad (1)$$

that is, the raw one-sided magnitude spectrum: no logarithmic compression, no per-spectrum renormalization, and the DC bin (index 0) is retained. With $L = 128$, this yields 65 bins per channel. Section 4.9 ablates the three natural alternatives—log-magnitude $\log(1 + |\cdot|)$, per-channel ℓ_2 spectral normalization, and dropping the DC bin—and reports how much the reported gain depends on this choice.

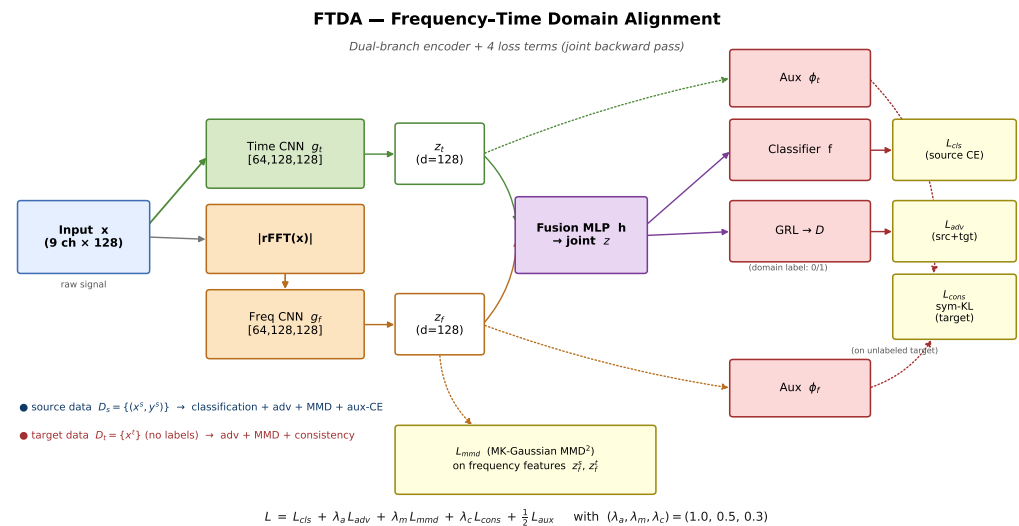


Figure 1. FTDA architecture. The window x and its magnitude spectrum $S(x)$ are encoded by two 1-D CNNs and fused into the joint feature z . The classifier f and the GRL-fed discriminator D read z ; the auxiliary heads ϕ_t, ϕ_f each read a single view; the frequency MMD is computed directly between z_f^s and z_f^t , bypassing the fusion. Loss placement follows Equation (2).

3.3. Losses

The model is optimized with the composite loss defined in Equation (2):

$$\mathcal{L} = \mathcal{L}_{cls}(s) + \lambda_a \mathcal{L}_{adv}(s, t) + \lambda_m \mathcal{L}_{mmd}^{freq}(s, t) + \lambda_c \mathcal{L}_{cons}^{tf}(t) + \lambda_x \mathcal{L}_{aux}(s). \quad (2)$$

The five terms of Equation (2) are detailed below.

(a) Classification loss. Standard cross-entropy on the source-domain joint features.

(b) Adversarial alignment. A domain discriminator D takes the joint feature z through a GRL with annealed coefficient $\lambda(p) = 2/(1 + e^{-10p}) - 1$, $p \in [0, 1]$, as in DANN [1].

(c) Frequency MMD. Multi-kernel Gaussian MMD² between source and target frequency features $g_f(x^s)$ and $g_f(x^t)$. Bandwidths are derived from the median heuristic with multipliers $\{1, 2, 4, 8\}$.

(d) Time–Frequency cross-view consistency. Two auxiliary linear classifiers ϕ_t, ϕ_f read the time-only and frequency-only features respectively. On unlabeled target x^t , we minimize the symmetric KL between their softmax predictions, $\mathcal{L}_{cons}^{tf} = \frac{1}{2}(\text{KL}(p_t \| p_f) + \text{KL}(p_f \| p_t))$.

(e) Auxiliary source supervision. The same two heads are additionally trained on the source labels, $\mathcal{L}_{aux}(s) = \text{CE}(\phi_t(g_t(x^s)), y^s) + \text{CE}(\phi_f(g_f(x^s)), y^s)$. This term is not cosmetic: it is what makes (d) a meaningful constraint. The consistency loss only measures agreement between ϕ_t and ϕ_f , so it is minimized perfectly whenever the two heads emit the same distribution—including the degenerate solution in which both collapse to a constant, label-independent output. Anchoring both heads to the source label space removes that shortcut. Section 4.8 verifies this empirically: with $\lambda_x = 0$, the two heads converge to the uniform distribution, and $\mathcal{L}_{cons}^{tf}$ collapses to $\approx 10^{-4}$, i.e., it is satisfied vacuously.

We fix $(\lambda_a, \lambda_m, \lambda_c, \lambda_x) = (1.0, 0.5, 0.3, 0.5)$ across all runs.

3.4. Training Algorithm

Algorithm 1 gives the full per-step training procedure that realizes Equation (2); all four losses are combined into a single backward pass.

Three design choices in Algorithm 1 are not forced by the formulation and are worth stating explicitly; the remaining, purely mechanical steps are documented in Appendix A.

Annealed adversarial pressure.

The GRL coefficient grows monotonically from 0 at $t = 0$ toward 1 as $t \rightarrow T$ rather than being held constant. Early in training, the encoder, therefore, concentrates on classification, and alignment pressure is introduced only once the features are discriminative enough to be aligned without destroying class structure.

MMD is placed on the frequency view, not on the joint feature.

The adversarial term already operates on z , so an MMD on the same representation would largely duplicate it. Attaching MMD to z_f instead supplies an alignment signal on the spectral view that the adversarial path does not directly exert. Section 4.7 quantifies how much this placement actually contributes.

The auxiliary heads are deliberately single-view.

ϕ_t and ϕ_f read z_t and z_f , respectively, whereas the main classifier f reads the fused z . This asymmetry is what gives the consistency term content: requiring two functionally distinct predictors to agree on the same unlabeled window is a non-trivial constraint, whereas agreement between two heads on the same input would be trivial. The symmetric KL keeps the constraint impartial between the two views, and the auxiliary supervision of Equation (2)(e) is what prevents it from being satisfied by mutual collapse.

Algorithm 1 FTDA training step.

Require: labeled source mini-batch (X_s, Y_s) ; unlabeled target mini-batch X_t ; global step t ; total steps T ; loss weights $\lambda_a, \lambda_m, \lambda_c, \lambda_x$ (defaults 1.0, 0.5, 0.3, 0.5)

Require: time encoder g_t , freq. encoder g_f , fusion MLP h ; main classifier f ; domain discriminator D ; auxiliary classifiers ϕ_t, ϕ_f

```

1:                                     ▷ Forward pass: both domains, both views
2:  $z_t^s \leftarrow g_t(X_s)$ ;  $z_f^s \leftarrow g_f(|\text{rFFT}(X_s)|)$ ;  $z^s \leftarrow h([z_t^s; z_f^s])$ 
3:  $z_t^t \leftarrow g_t(X_t)$ ;  $z_f^t \leftarrow g_f(|\text{rFFT}(X_t)|)$ ;  $z^t \leftarrow h([z_t^t; z_f^t])$ 
4:                                     ▷ (a) Classification on labeled source
5:  $\mathcal{L}_{\text{cls}} \leftarrow \text{CE}(f(z^s), Y_s)$ 
6:                                     ▷ (b) Adversarial alignment via Gradient Reversal
7:  $\lambda_{\text{GRL}} \leftarrow 2/(1 + e^{-10t/T}) - 1$ 
8:  $d_s \leftarrow D(\text{GRL}(z^s, \lambda_{\text{GRL}}))$ ;  $d_t \leftarrow D(\text{GRL}(z^t, \lambda_{\text{GRL}}))$ 
9:  $\mathcal{L}_{\text{adv}} \leftarrow \text{CE}([d_s; d_t], [\mathbf{0}; \mathbf{1}])$ 
10:                                     ▷ (c) Multi-Kernel MMD on frequency features
11:  $\mathcal{L}_{\text{mmd}} \leftarrow \text{MK-Gauss-MMD}^2(z_f^s, z_f^t; \text{ kernels } = \{1, 2, 4, 8\} \cdot \sigma_{\text{med}})$ 
12:                                     ▷ (d) Cross-view consistency on unlabeled target
13:  $p_t \leftarrow \text{softmax}(\phi_t(z_t^t))$ ;  $p_f \leftarrow \text{softmax}(\phi_f(z_f^t))$ 
14:  $\mathcal{L}_{\text{cons}} \leftarrow \frac{1}{2}(\text{KL}(p_t \| p_f) + \text{KL}(p_f \| p_t))$ 
15:                                     ▷ (e) Auxiliary supervision keeps  $\phi_t, \phi_f$  source-discriminative
16:  $\mathcal{L}_{\text{aux}} \leftarrow \text{CE}(\phi_t(z_t^s), Y_s) + \text{CE}(\phi_f(z_f^s), Y_s)$ 
17:                                     ▷ Composite loss and update
18:  $\mathcal{L} \leftarrow \mathcal{L}_{\text{cls}} + \lambda_a \mathcal{L}_{\text{adv}} + \lambda_m \mathcal{L}_{\text{mmd}} + \lambda_c \mathcal{L}_{\text{cons}} + \lambda_x \mathcal{L}_{\text{aux}}$ 
19: Backprop on  $\mathcal{L}$ ; one Adam step

```

3.5. Optimization

Adam (lr 10^{-3} , weight decay 10^{-4}), cosine schedule, batch size 64, and 30 epochs. The main method comparison uses 10 seeds $\{0, \dots, 9\}$ per condition; the ablations and sensitivity studies, which compare variants of a single method rather than establishing headline claims, use three seeds $\{0, 1, 2\}$. Every table states which applies. Final model selection is carried out by the best source-validation accuracy and never uses target labels.

3.6. Dataset and Transfer Tasks

UCI HAR [6] contains 30 subjects, 6 activities, and 10,299 windows of 128 samples (2.56 s, 50% overlap) at 50 Hz from 9 inertial channels. We define four cross-subject transfer tasks (Table 2).

Table 2. Cross-subject transfer tasks used throughout the experiments.

Task	Source Subjects (Count)	Target Subjects (Count)
T1	11	6
T2	10	3
T3	15	15
T4	15	15

T3 and T4 are mirror tasks (source/target swapped) and stress generalization in both directions across a 15×15 partition.

3.7. Baselines

- Source-only: Identical backbone trained on source labels only.
- DANN [1]: GRL + binary domain discriminator on the joint feature, with no MMD and no consistency.
- CoDATS-style: Adversarial + MMD on the joint feature (no frequency branch and no consistency).
- Time-Wide-DANN (parameter-increased control): Time-only encoder widened to channels [96, 192, 192] (0.25 M parameters, 5.41 M MACs) trained with the DANN loss.
- Time-Wide-MAC-DANN (parameter- and MAC-matched control): The same encoder widened to [112, 224, 224] (0.332 M parameters, 7.26 M MACs, against FTDA's 0.308 M and 7.55 M) trained with the DANN loss.

All methods share the time-branch backbone, optimizer, schedule, batch size, and GRL warm-up; only the loss configuration changes.

3.8. Evaluation Protocol

The evaluation protocol follows the conventions of the ADATIME benchmarking suite [22]. We report target-domain accuracy and macro-F1 as the mean $\pm$ standard deviation over 10 seeds, together with bootstrap 95% confidence intervals over the resulting 40 (task, seed) observations. Statistical comparisons use paired *t*-tests and Wilcoxon signed-rank tests over those pairs. We additionally report per-class precision/recall/F1 and a proxy $\mathcal{A}$ -distance [2] estimated by 2-fold logistic regression on the joint features.

Splitting, and a leakage hazard we corrected.

All source/target partitions are by subject: a subject's windows appear in exactly one domain, so no window is shared across the domain boundary. This matters because UCI HAR windows overlap by 50%, and we verified this directly on the released data—the second half of window *i* is bit-identical to the first half of window *i* + 1 (Pearson *r* = 1.000).

That same overlap created a hazard inside the source domain, which we have corrected in this version. The source train/validation split was previously drawn at the window level, so two windows sharing half their samples could fall on opposite sides of the split. Since model selection uses source-validation accuracy, this inflated the selection signal. Table 3 quantifies the effect: source-validation accuracy drops by 4.01 pp once the split is made by subject, confirming the optimism, while target accuracy—the number the paper actually reports—moves by only −0.41 pp because the target subjects were disjointed under both regimes. All results in this paper use the corrected subject-level source-validation split.

Table 3. Effect of correcting the source train/validation split. FTDA, with 4 tasks $\times$ 3 seeds. The leakage inflated only the model-selection signal; the reported target metric is essentially unchanged.

Source Train/Val Split	Source-Val Accuracy (%)	Target Accuracy (%)
Window-level (previous)	98.17	95.49
Subject-level (corrected, used here)	94.16	95.08
Difference	−4.01	−0.41

Use of target data.

Unlabeled target windows are used during adaptation, as is standard in unsupervised domain adaptation, and the same windows are used for evaluation. No target label is read at any point during training or model selection. We state this explicitly because the transductive setting is not universal in this literature.

4. Results

4.1. Main Comparison

Table 4 reports the target-domain accuracy per task; Figures 2 and 3 visualize accuracy and macro-F1 respectively. In both figures, FTDA sits above every baseline on all four transfer tasks. The margin over the strongest competing method is widest on T1 (+3.48 pp over Freq-only-DANN) and narrowest on T4 (+0.51 pp over RAINCOAT-style); T2 is compressed at the top because Source-only already reaches 97.67% there, leaving little headroom. The error bars, one standard deviation across 10 random seeds, are small in absolute terms for every method, so the ordering is not a seed artifact: the time-only baselines overlap one another, whereas FTDA lies outside their spread on every task. Macro-F1 (Figure 3) tracks accuracy closely, indicating that FTDA's gain is not driven by improvements on a single dominant class but is distributed across the six activities; per-class behavior is examined further in Section 4.14.

Table 4. Target-domain accuracy (%) per task; mean $\pm$ std over 10 seeds under the corrected subject-level validation split. The last row is an oracle trained with target labels and is not a competing method.

Method	Target Accuracy (%)				
	T1	T2	T3	T4	Mean
Source-only	87.02 $\pm$ 1.41	97.67 $\pm$ 0.56	93.79 $\pm$ 0.37	91.34 $\pm$ 0.55	92.46 $\pm$ 3.99
DANN	86.98 $\pm$ 0.70	97.64 $\pm$ 0.95	93.94 $\pm$ 0.71	90.62 $\pm$ 0.77	92.29 $\pm$ 4.07
CoDATS-style	86.75 $\pm$ 0.82	97.68 $\pm$ 0.97	94.04 $\pm$ 0.55	90.78 $\pm$ 0.73	92.31 $\pm$ 4.15
Freq-only-DANN	88.00 $\pm$ 2.68	93.87 $\pm$ 1.61	89.69 $\pm$ 1.60	90.78 $\pm$ 0.45	90.58 $\pm$ 2.75
TF-C-style	86.57 $\pm$ 1.10	98.31 $\pm$ 1.07	94.79 $\pm$ 0.51	92.31 $\pm$ 0.73	92.99 $\pm$ 4.41
RAINCOAT-style	87.94 $\pm$ 0.94	97.97 $\pm$ 0.85	94.66 $\pm$ 0.67	92.70 $\pm$ 0.86	93.32 $\pm$ 3.77
FTDA	91.48 $\pm$ 0.96	99.56 $\pm$ 0.38	95.78 $\pm$ 0.33	93.21 $\pm$ 1.50	95.01 $\pm$ 3.21
Target-only (oracle)	93.53 $\pm$ 0.77	97.01 $\pm$ 2.49	97.00 $\pm$ 0.48	95.30 $\pm$ 0.61	95.71 $\pm$ 1.96

A note on statistical power.

An earlier version of this work used three seeds ($n = 12$) and reported the comparison against RAINCOAT-style as *not* significant (+0.36 pp, $p = 0.275$). Raising the seed count to 10 and correcting the validation split moved the point estimate to +1.69 pp. We then pooled the 40 (task, seed) runs as independent observations and obtained $p = 2.0 \times 10^{-7}$, which we reported. That pooling was wrong: the four tasks draw on overlapping subject pools, and T3/T4 are mirrors of one another, so the runs are not independent and the test was anti-conservative. Averaging to four task means does not repair this, as the task

means inherit the same shared subjects. Therefore, Table 5 reports the task-level differences descriptively, without a global test, and confines inference to the within-task comparisons, under which FTDA and RAINCOAT-style are separated on three of the four tasks. We record the full sequence rather than only the final number because the conclusion we drew at the intermediate step was an artifact of the analysis and not of the data.

Table 5. Target accuracy, with task-level differences reported descriptively. The four transfer tasks share subjects, and T3/T4 are mirror configurations, so even the four task means are not independent observations; we therefore attach no global significance test to them. The Δ and interval columns summarize the four task-mean differences (mean and 95% *t*-interval) and are descriptive only. Inference is confined to the “per-task” column: within a task, the 10 seeds are exchangeable, and the entry counts how many of the four within-task paired tests survive Holm correction.

Method	Mean (%)	Δ vs. FTDA (pp)	Task-Level Interval	Per-Task (Holm)
Source-only	92.46	+2.55	[+0.53, +4.57]	4/4
DANN	92.29	+2.71	[+0.75, +4.68]	4/4
CoDATS-style	92.31	+2.69	[+0.48, +4.90]	4/4
Freq-only-DANN	90.58	+4.43	[+1.62, +7.23]	4/4
TF-C-style	92.99	+2.01	[−1.06, +5.09]	3/4
RAINCOAT-style	93.32	+1.69	[−0.40, +3.78]	3/4
FTDA	95.01	—	—	—
Target-only (reference)	95.71	−0.70	—	—

Within every task, FTDA is ahead of all four single-view baselines (4/4 after Holm correction). Against the two dual-view competitors, the within-task tests separate FTDA on three of four tasks, with T4 being the exception, and the descriptive task-level intervals include zero; no claim of superiority over the dual-view methods is made. An earlier version of this table pooled the 40 runs as being independent and reported $p < 10^{-6}$ throughout, and an intermediate version attached a paired test to the four task means; both treatments understated the dependence between runs, and neither is retained. The last row is a target-only model trained and evaluated on the same target windows: it measures fit rather than generalization and is not a valid ceiling, and no parity claim is made from it.

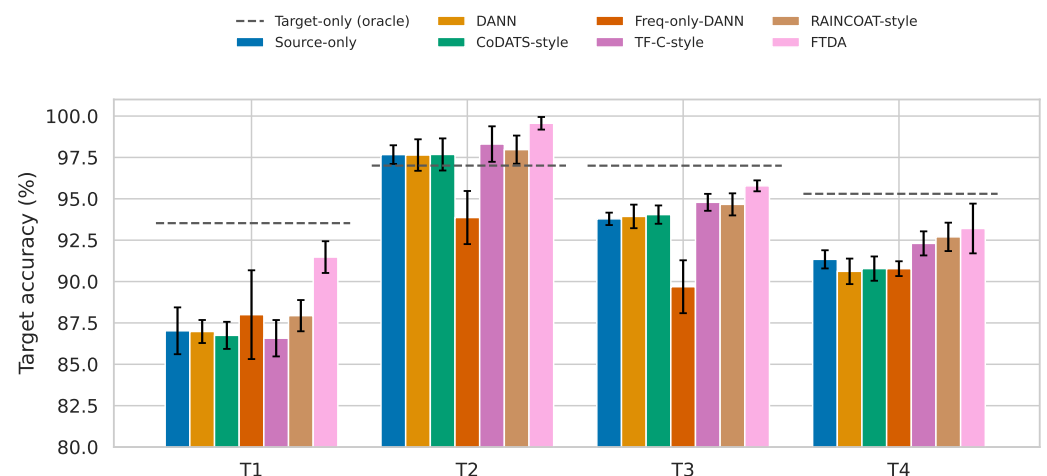


Figure 2. Target-domain accuracy on the four UCI HAR cross-subject transfer tasks. Each marker is the mean over 10 random seeds and the bar through it spans ± 1 standard deviation; methods are ordered by their overall mean, and FTDA is drawn in color. The dashed vertical line marks the target-supervised oracle for that task. The horizontal scale differs between panels because the four tasks differ in difficulty.

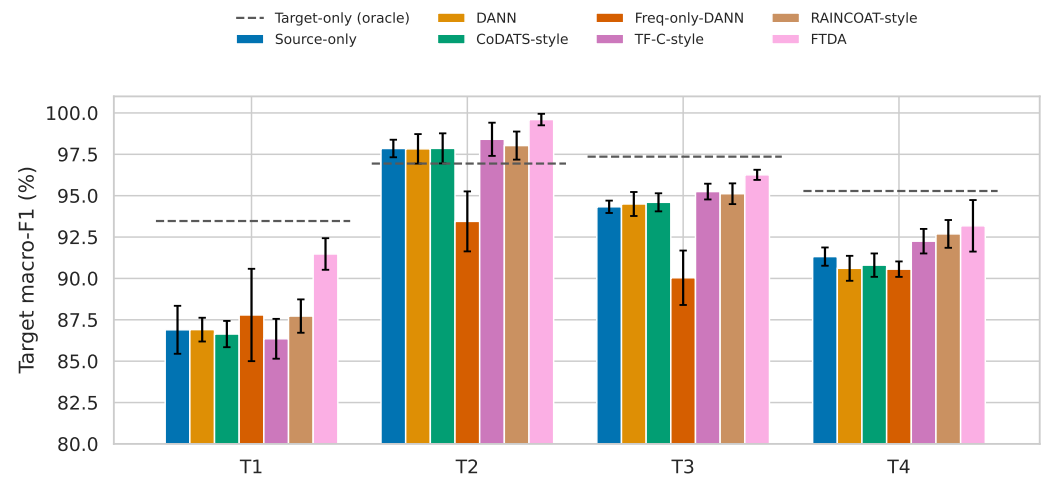


Figure 3. Target-domain macro-F1 on the same four tasks, drawn as in Figure 2.

4.2. Confusion Matrices

Figure 4 shows row-normalized confusion matrices of FTDA on T1–T4. Errors concentrate among the static postures (SITTING / STANDING / LAYING) when subject-specific posture priors differ, and among the walking variants when gait cadence shifts.

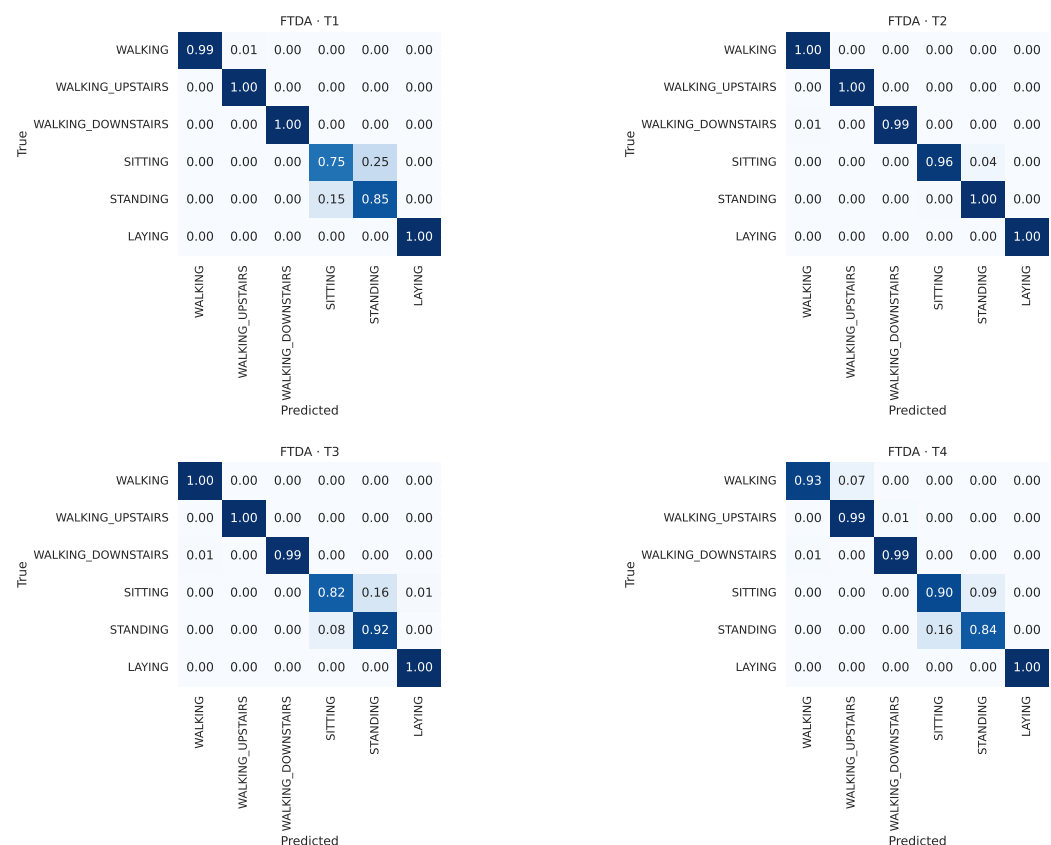


Figure 4. Row-normalized confusion matrices of FTDA on the target domain for tasks T1–T4 (seed 0). Darker blue indicates a larger fraction of the row; each row sums to one.

4.3. Feature-Space Inspection

Figure 5 shows the t-SNE embedding of source and target features under Source-only and FTDA for T1. FTDA produces visibly tighter intra-class clusters and closer source–target alignment of the same class.

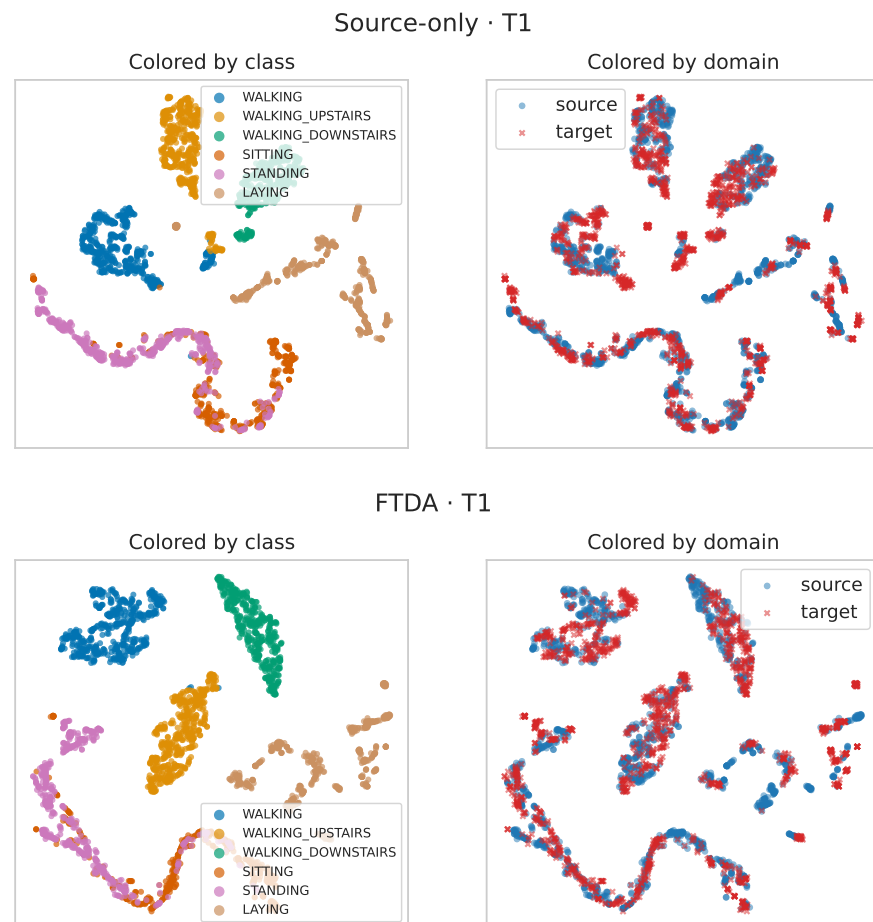


Figure 5. t-SNE of the joint feature on T1. (**Top**) Source-only. (**Bottom**) FTDA. Right panel of each pair denotes domain.

4.4. Training Dynamics

Figure 6 shows the loss components and source-validation/target-accuracy trajectories. Classification and adversarial losses decrease monotonically; the frequency-MMD and consistency losses settle to small positive values, indicating the encoder reaches a near-equilibrium between discriminability and alignment.

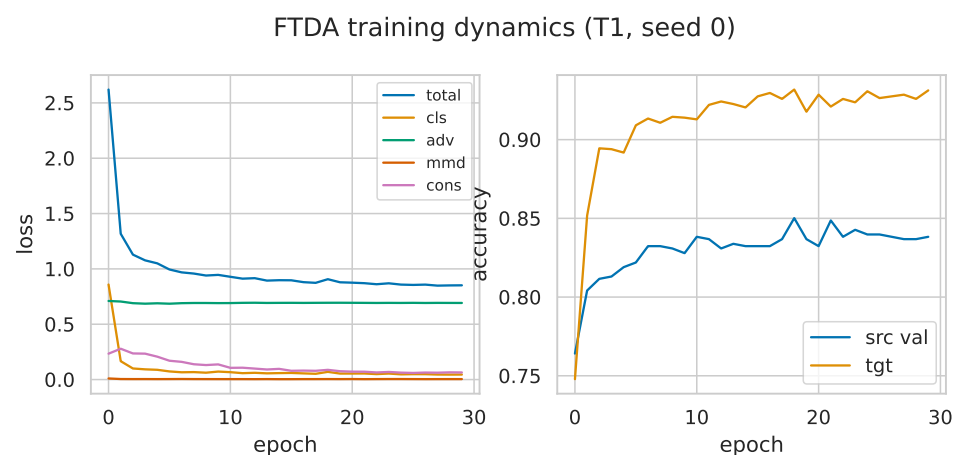


Figure 6. Training dynamics of FTDA on T1, seed 0.

4.5. Frequency-Domain Shift

Figure 7 plots channel-wise average FFT magnitudes for source and target subjects on T1, revealing differences in the 1–3 Hz gait band that motivate the frequency branch.

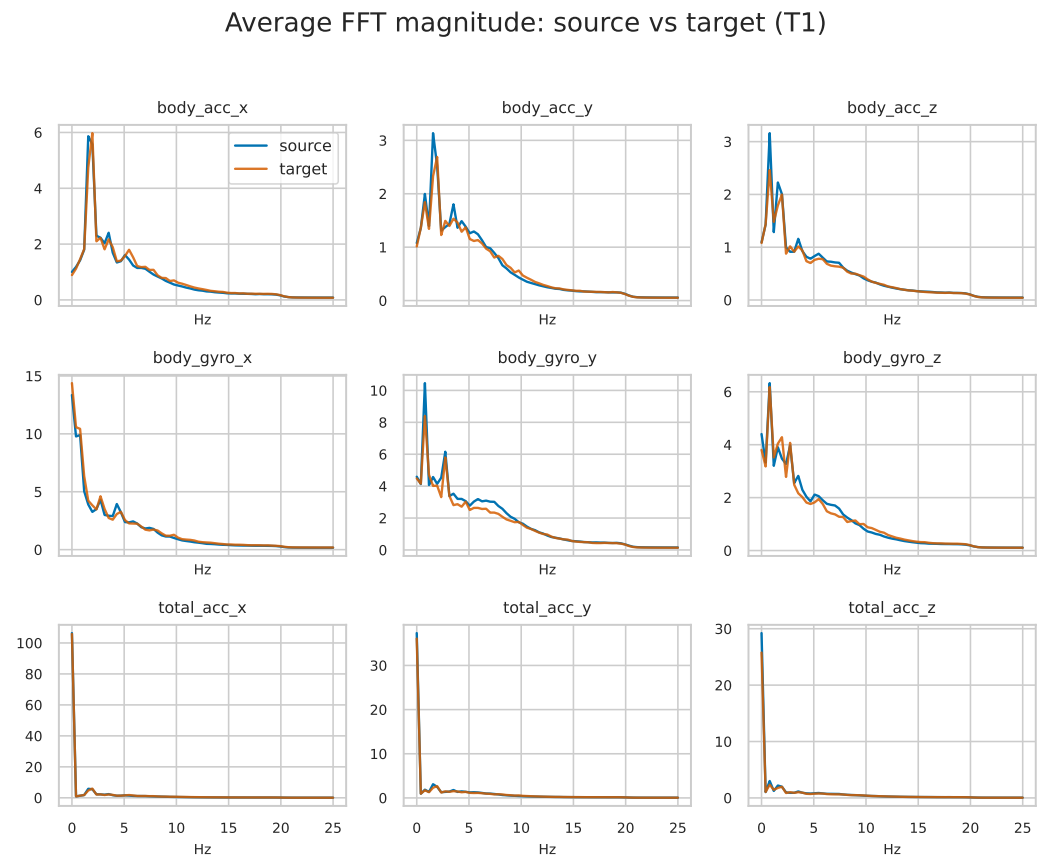


Figure 7. Source vs. target average FFT magnitude on T1, per channel.

4.6. Capacity-Matched Control

To rule out a capacity confound, we evaluate two widened time-only controls (Table 6). The controls were trained on seeds $\{0, 1, 2\}$, so every entry in the table—including DANN and FTDA—was recomputed over those same three seeds, and all comparisons are paired over identical runs. Time-Wide-DANN widens the encoder to channels $[96, 192, 192]$ (0.25 M parameters, 5.41 M MACs); Time-Wide-MAC-DANN, with channels $[112, 224, 224]$, uses 0.332 M parameters and 7.26 M MACs—more parameters than FTDA and MACs, within 4% of it. Against DANN, the paired differences are +0.35 pp (95% CI $[-0.43, +1.12]$) for the first control and +0.29 pp (95% CI $[-0.59, +1.16]$) for the second; FTDA gains +2.97 pp (95% CI $[+1.57, +4.38]$) on the same runs. Neither control’s interval excludes zero. Added capacity alone, therefore, does not appear to explain the observed improvement (Figure 8).

Table 6. Capacity controls. Time-Wide-DANN matches FTDA on parameters only; Time-Wide-MAC-DANN matches it on parameters and MACs (exceeding FTDA on parameters). All four rows—including DANN and FTDA—are computed over the same seeds $\{0, 1, 2\}$, so every comparison is paired over identical runs.

Method	Params (M)	MACs (M)	Target Accuracy (%)				
			T1	T2	T3	T4	Mean
DANN	0.15	2.5	86.06	98.06	93.47	90.82	92.10
Time-Wide-DANN	0.25	5.4	86.76	97.51	94.58	90.95	92.45

Table 6. Cont.

Method	Params (M)	MACs (M)	Target Accuracy (%)				
			T1	T2	T3	T4	Mean
Time-Wide-MAC-DANN	0.33	7.3	87.07	97.57	94.16	90.76	92.39
FTDA	0.31	7.5	92.17	99.58	95.48	93.08	95.08

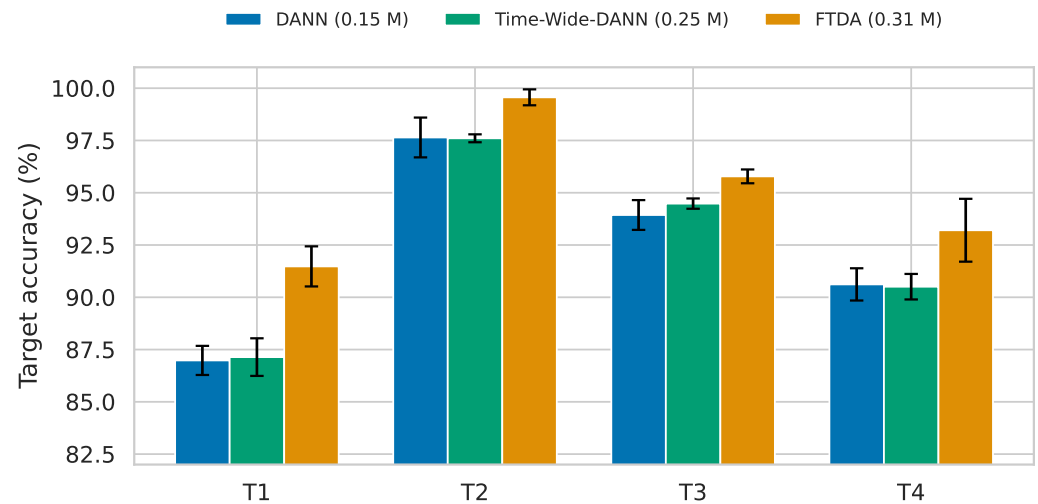


Figure 8. Capacity controls. Widening the time-only encoder leaves it close to DANN on every task, whereas FTDA is well above all three. Markers are means with ± 1 standard deviation; the horizontal scale differs between panels.

4.7. Component Ablation

Table 7 and Figure 9 quantify the marginal contribution of each FTDA component, with each row adding one component to the row above.

Table 7. Component ablation of FTDA. Mean target accuracy ($\pm$ std) over 4 tasks $\times$ 3 seeds, with corrected validation split. Each row adds one component to the row above.

Variant	Target Accuracy (%)	Target Macro-F1 (%)	Δ (pp)
time_only	92.10 $\pm$ 4.61	92.26	—
time + freq_concat	93.63 $\pm$ 4.73	93.73	+1.53
+ freq-MMD	93.06 $\pm$ 4.80	93.22	−0.57
+ consistency (FTDA)	95.08 $\pm$ 3.05	95.20	+2.02

Three things follow, and the second is unfavorable to a component of our own design.

First, adding the frequency branch as a concatenated feature is worth +1.53 pp on its own. Second, for the frequency MMD term, no measurable benefit was detected: the paired change is −0.57 pp with a 95% CI of $[-1.28, +0.13]$, which includes zero; the λ_m sweep in Section 4.13 is consistent with this, being flat to slightly favorable at $\lambda_m = 0$. We report this plainly: on this benchmark, the MMD term can be removed at no cost, and a practitioner reproducing FTDA should feel free to drop it. We retain it in the reported configuration only because the configuration was fixed before any target-domain measurement, and we did not want to re-select components after seeing target numbers. Third, and largest, the cross-view consistency term is worth +2.02 pp.

The ordering matters for how the method should be understood. The frequency branch is not primarily useful as extra features to concatenate; it is useful because it creates a second view against which the target predictions can be held consistent. The two components

are, therefore, not independent additions—the consistency term cannot exist without the second view—and together they account for +2.98 pp of the +2.98 pp total, with the MMD term contributing negatively.

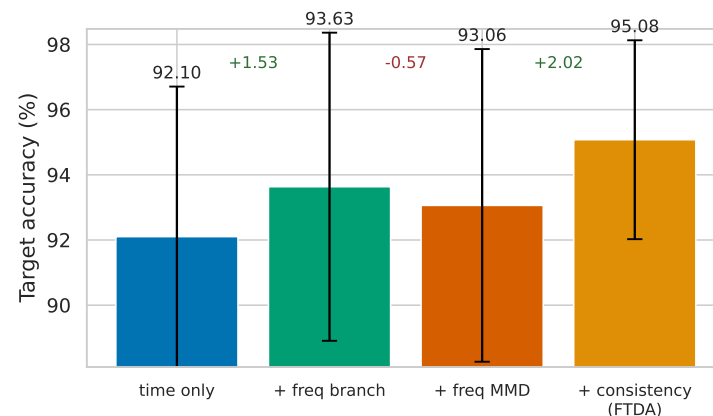


Figure 9. Component ablation of FTDA over 4 tasks $\times$ 3 seeds. Left: Cumulative target accuracy as each component is added. Right: The change contributed by each step, computed within (task, seed) pairs and shown with its 95% confidence interval, with the vertical line marking no change. The paired view is the one on which the argument rests: the spread in the left panel is dominated by how much the four tasks differ from one another rather than by the components themselves.

4.8. Necessity of the Auxiliary Supervision Term

Equation (2)(e) anchors the two view-specific heads to the source label space. Table 8 sweeps its weight λ_x and, alongside accuracy, reports the converged value of the consistency loss and the mean predictive entropy of each auxiliary head.

Table 8. Effect of the auxiliary source-supervision weight λ_x . Mean over 4 tasks $\times$ 3 seeds. $H(\phi)$ is the mean predictive entropy in nats; $\log 6 = 1.792$ is the entropy of the uniform distribution over the six classes.

λ_x	Target Accuracy (%)	$\mathcal{L}_{\text{cons}}^{\text{tf}}$	$H(\phi_t)$	$H(\phi_f)$
0 (term removed)	93.25 ± 4.55	0.0000	1.7916	1.7916
0.25	94.99 ± 3.28	0.0521	0.1776	0.1800
0.5 (default)	95.08 ± 3.05	0.0713	0.1556	0.1534
1.0	94.91 ± 3.29	0.0998	0.1415	0.1392

The mechanism predicted in Section 3.3 is confirmed exactly. At $\lambda_x = 0$, both heads converge to an entropy of 1.7916 nats against a uniform-collapse reference of $\log 6 = 1.7918$: they emit the uniform distribution regardless of input. The consistency loss is correspondingly 0.0000—not small, but identically satisfied because the symmetric KL between two identical constant distributions is zero. In that regime, $\mathcal{L}_{\text{cons}}^{\text{tf}}$ contributes no gradient, and FTDA degenerates to a dual-branch encoder with adversarial and MMD alignment only; consistent with this, its accuracy (93.25%) matches the “+ freq-MMD” ablation row of Table 7 (93.06%) to within noise. Any $\lambda_x > 0$ restores discriminative heads (entropy 0.14–0.18) and a non-vacuous consistency term.

Removing the term costs 1.83 pp, which is close to the 2.02 pp that Table 7 attributes to the consistency component—as it must be, since the two are the same mechanism seen from two directions. The auxiliary term is, therefore, best described as enabling $\mathcal{L}_{\text{cons}}^{\text{tf}}$ rather than as an independent source of gain: the pair $(\mathcal{L}_{\text{cons}}^{\text{tf}}, \mathcal{L}_{\text{aux}})$ must be adopted or dropped together. Setting λ_x anywhere in $[0.25, 1.0]$ gives results within 0.2 pp of each other, so the exact value is not critical.

4.9. Spectral Preprocessing

Table 9 ablates the magnitude-spectrum transform defined in Equation (1).

Table 9. Effect of the spectral preprocessing applied to the frequency branch. Mean over 4 tasks $\times$ 3 seeds.

Spectral Input	Target Accuracy (%)	Target Macro-F1 (%)	Δ (pp)
raw magnitude (used throughout)	95.08 ± 3.05	95.20	—
log magnitude $\log(1 + \cdot)$	95.94 ± 2.19	96.03	+0.86
ℓ_2 spectral normalization	94.96 ± 2.94	95.07	−0.12
DC bin removed	94.81 ± 2.73	94.93	−0.27

Two findings are worth stating plainly. First, removing the DC bin hurts (−0.27 pp). The DC bin of a window encodes its mean offset, which, on the accelerometer channels, reflects device orientation relative to gravity and is, thus, partly subject-specific; the ablation indicates that its discriminative value for the static postures outweighs that domain specificity, so it should be kept rather than discarded. Second, logarithmic compression helps (+0.86 pp, with the cross-task standard deviation falling from 3.05 to 2.19). A plausible reading—an interpretation, not a demonstrated mechanism—is that the raw magnitude spectrum is dominated by high-power low-frequency bins and that log compression rebalances the representation toward the mid-band where gait cadence lies. Taken together, compress the spectrum; do not truncate it.

We nonetheless report all other results with the raw magnitude because that configuration was fixed before any target-domain measurement. Adopting log-magnitude now, on the strength of these target-domain numbers, would be a selection on the target and would violate the label-free model-selection protocol used everywhere else in this paper. We therefore record the +0.86 pp as a reproducible ablation finding and an immediate direction for future work rather than folding it into the headline result.

4.10. Comparison with Frequency-Aware Baselines

The submitted version compared only against time-domain baselines. Three frequency-aware competitors are included in Tables 4 and 5 under the identical protocol: a frequency-only DANN, a RAINCOAT-style baseline whose alignment term is an entropic-OT divergence, and a TF-C-style baseline built on time–frequency contrastive pre-training. Table 10 states what each re-implementation does and does not reproduce.

Table 10. Scope of the two re-implementations. We label them “-style” because they reproduce the mechanism each paper contributes to our setting, not the full published system.

	Reproduced	Omitted
RAINCOAT-style [4]	Dual time–frequency encoder; Sinkhorn (entropic-OT) divergence as the alignment objective, replacing GRL/MMD	The universal-DA detection and correction stage (our setting is closed-set, so it is in-applicable); the original backbone and augmentation details
TF-C-style [5]	The two-stage protocol: time–frequency contrastive pre-training on unlabeled source+target windows, then supervised source fine-tuning	The specific augmentation family and the frequency-perturbation scheme of the original

Three observations follow.

The frequency view alone is not sufficient. Freq-only-DANN is the weakest method in the whole study (90.58%), below every time-only baseline. The spectral view is a complement to the temporal view, not a replacement for it, which is the premise the dual-branch design rests on.

Both dual-view competitors improve on the single-view baselines, but not equally. RAINCOAT-style (93.32%) and TF-C-style (92.99%) sit above the time-only methods (92.29–92.46%), and FTDA (95.01%) sits above both. The three differ in how they exploit the second view—entropic OT alignment, contrastive pre-training, and target-side prediction consistency, respectively—and the ordering suggests that how the second view is used matters, not merely that it is present.

FTDA's point estimate is +1.69 pp above RAINCOAT-style, with a descriptive task-level interval of $[-0.40, +3.78]$ pp, which includes zero; the within-task tests separate the two methods on three of the four tasks, with T4 being the exception. The defensible claim is, therefore, that FTDA is ahead of every single-view baseline and is at least competitive with the dual-view ones, not that it is better than them. FTDA also requires no optimal-transport solver in the training loop, which is a practical rather than an accuracy advantage.

4.11. Leave-One-Subject-Out Evaluation

The four fixed transfer tasks average over groups of target subjects, which can mask how the method behaves on an individual it happens to handle badly. We therefore add a leave-one-subject-out protocol: for each subject in a pool of 10, that subject alone is the target domain, and the remaining nine are the source. Table 11 reports the mean, the between-subject standard deviation, and the worst single subject, and Figure 10 shows the full per-subject distribution.

Table 11. Leave-one-subject-out on UCI HAR (subjects 1–10, 3 seeds each). The spread across subjects is as informative as the mean.

Method	LOSO Mean (%)	Std Across Subjects (pp)	Worst Subject (%)
Source-only	89.16	9.71	58.50
DANN	91.01	8.47	66.33
RAINCOAT-style	86.43	9.60	60.54
FTDA	92.73	4.31	82.99

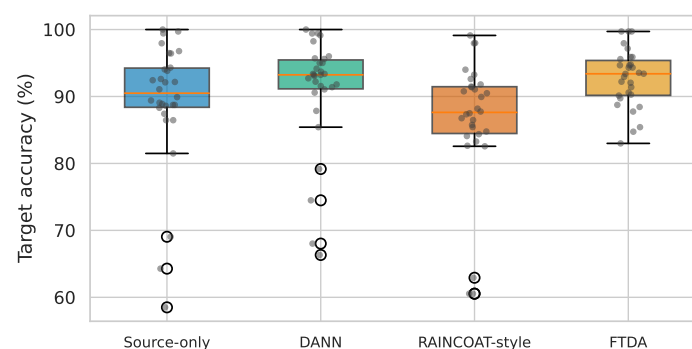


Figure 10. Leave-one-subject-out accuracy on UCI HAR. Each point is one held-out subject over three seeds, and the box spans the quartiles; the short rule below each box marks that method's worst subject, with the value printed beside it. FTDA's advantage lies as much in lifting the worst case from 58.5–66.3% to 83.0% as in the median.

The mean ordering is consistent with the fixed-partition results, but the variance is the more useful number. FTDA's between-subject standard deviation is roughly half

that of every other method, and its worst subject is handled at 82.99% against 58–66% elsewhere. Since a deployment is judged by the users, it fails rather than by the average user, we regard this as the more operationally meaningful comparison. RAINCOAT-style, which is competitive on the fixed partition, is the weakest method here, indicating that its optimal-transport alignment is less stable when the target domain is a single individual.

4.12. A Second Dataset and a Different Kind of Shift

Every result above holds the recording device fixed and varies the subjects. To test whether the frequency view helps under a different kind of shift, we added HHAR [23], in which the same nine subjects carried four phone models (nexus4, s3, s3mini, and samsungold) simultaneously. Fixing the subjects and varying the device isolates hardware heterogeneity—sampling-rate jitter, axis calibration, and noise floor—from subject variation. Signals were resampled from each device’s native rate to a common value of 50 Hz and were windowed identically to UCI HAR (128 samples, 50% overlap), giving 76,647 windows across four transfer pairs. Table 12 reports the aggregate result, and Figure 11 reports the per-pair breakdown.

Table 12. Cross-device transfer on HHAR (4 device pairs $\times$ 3 seeds). All four pairs are formed from the same nine subjects, so the 12 (pair, seed) runs are not independent; the p values (paired t -test against FTDA) are exploratory. n.s.: not significant.

Method	Target Accuracy (%)	Δ vs. FTDA	p
Source-only	89.83 ± 5.40	+5.07	6.0×10^{-4}
TF-C-style	89.74 ± 6.27	+5.16	3.2×10^{-3}
RAINCOAT-style	91.16 ± 4.47	+3.74	2.2×10^{-3}
DANN	94.42 ± 3.85	+0.48	0.54 (n.s.)
CoDATS-style	94.91 ± 2.99	−0.01	0.99 (n.s.)
FTDA	94.90 ± 2.05	—	—

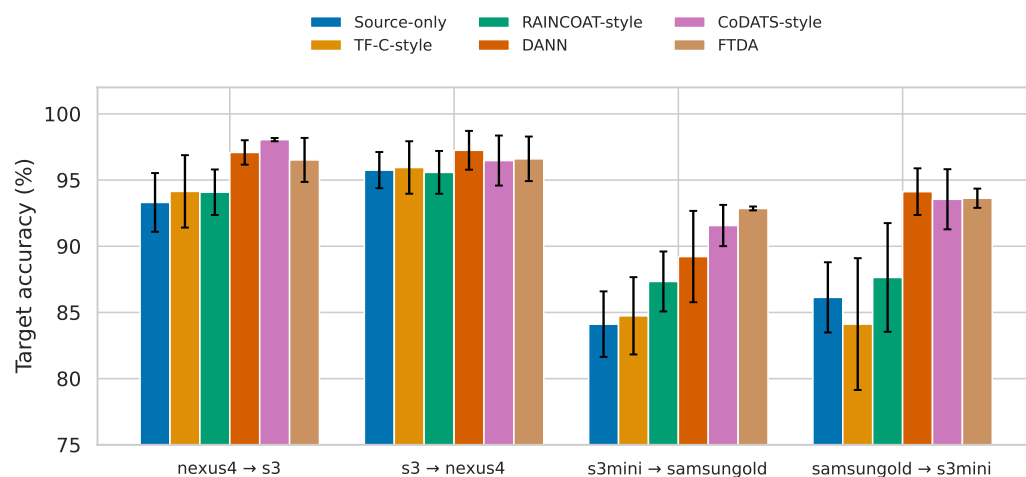


Figure 11. Cross-device transfer on HHAR, per device pair, drawn as in Figure 2. FTDA overlaps the time-only DANN and CoDATS-style baselines on every pair, in contrast to the cross-subject setting. The horizontal scale differs between panels.

The result does not replicate the cross-subject ordering, and we report it as found. FTDA is 5.07 pp above Source-only ($p = 0.001$), but we detect no difference from the two time-only adversarial baselines: the paired difference is +0.48 pp against DANN (95% CI $[-1.20, +2.16]$, $p = 0.54$) and −0.01 pp against CoDATS-style (95% CI $[-1.08, +1.07]$, $p = 0.99$). We stress what this does and does not license. With $n = 12$ paired observations, these intervals still admit differences of roughly two points in either direction—and because

all four device pairs are formed from the same nine subjects, the 12 runs are not independent, so we label these intervals exploratory rather than confirmatory; therefore, the correct statement is that the cross-subject advantage did not reproduce here, not that the methods are equivalent; establishing equivalence would require a pre-specified margin and a study powered for it, which we have not run.

Two readings are compatible with this result, and our design cannot separate them. The first is that the two shifts are not the same kind of object: cross-subject shift is largely a difference in gait cadence, a change in the shape of the spectrum, and, therefore, what a frequency branch is positioned to exploit, whereas axis-calibration offsets are not. The second is that our own preprocessing removed the spectral part of the device shift—resampling every stream to a common 50 Hz value is itself a spectral normalization, and sampling-rate differences are precisely a frequency-axis phenomenon. We cannot distinguish these from a single dataset, so we state the scope condition as a limitation rather than as an established mechanism: the cross-subject benefit of the spectral view did not transfer to the cross-device setting we tested.

4.13. Hyper-Parameter Sensitivity

Figure 12 sweeps λ_m and λ_c on T1 across three seeds. Target accuracy varies by less than ± 1 pp across an order of magnitude in either weight; the default $(\lambda_m, \lambda_c) = (0.5, 0.3)$ used throughout sits within the flat region of both curves.

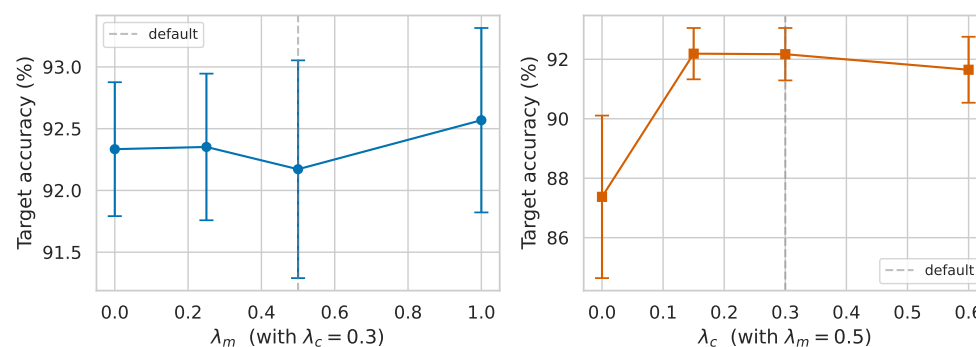


Figure 12. Sensitivity of T1 target accuracy to λ_m (left) and λ_c (right). Dashed lines mark the values used in the main experiments.

4.14. Per-Class Behavior and Domain Distance

Figure 13 shows per-class F1 of FTDA. Errors concentrate on the static postures SITTING and STANDING. Table 13 reports the proxy $\mathcal{A}$ -distance between source and target features on T1.

WALKING	0.995	0.997	0.996	0.961
WALKING_UPSTAIRS	0.997	1.000	0.999	0.956
WALKING_DOWNSTAIRS	0.998	0.997	0.997	0.990
SITTING	0.785	0.979	0.863	0.871
STANDING	0.814	0.982	0.890	0.872
LAYING	1.000	1.000	0.993	0.999
	T1	T2	T3	T4
Task				

Figure 13. Per-class F1 of FTDA on the target domain across the four transfer tasks (seed 0). Darker blue indicates higher F1; the printed value is shown in every cell.

Table 13. Proxy $\mathcal{A}$ -distance on T1. Lower $\mathcal{A}$ -distance means features are harder to tell apart as source/target ($\downarrow$: lower is better; $\uparrow$: higher is better).

Method	$\mathcal{A}$ -Distance $\downarrow$	Domain-Clf Error $\uparrow$
Source-only	0.82	0.294
DANN	0.85	0.288
CoDATS	0.86	0.286
FTDA	0.82	0.294

The proxy $\mathcal{A}$ -distance is similar across methods, while target accuracy varies by up to 4.4 pp; FTDA's gain does not come from reducing feature-level domain discriminability but from learning a more class-discriminative representation.

4.15. Computational Footprint

Table 14 lists the cost of every method. FTDA's parameter count is $\sim 2.1 \times$ DANN's, and its MAC count is $\sim 3.0 \times$. The absolute footprint is 0.31 M parameters and 7.55 M MAC per 2.56 s window. We report these counts rather than a deployment verdict: no on-device timing or energy measurement was made, so we cannot say what this costs on any particular wearable inference chip. The counts are given so that the cost of the second branch relative to a time-only encoder ($\sim 3.0 \times$ the MACs of DANN) is explicit.

Table 14. Parameter and computational footprint. MACs are per 2.56 s window.

Method	Parameters	MACs/Window
Source-only	0.129 M	2.52 M
DANN	0.146 M	2.52 M
CoDATS	0.146 M	2.52 M
Time-Wide-DANN	0.253 M	5.41 M
Time-Wide-MAC-DANN	0.332 M	7.26 M
FTDA	0.308 M	7.55 M

5. Discussion

Where the gain comes from.

Aggregated over four tasks and 10 seeds, FTDA reaches 95.01% against 92.46% for Source-only (+2.55 pp) and 92.29/92.31% for DANN and CoDATS-style (+2.71/+2.69 pp). The gain is largest on T1 (+4.46 pp over Source-only) and smallest on T2 (+1.89 pp), where Source-only already reaches 97.7%, and little headroom remains.

Decomposing the contribution, and a component that does not earn its place.

Table 7 gives a breakdown that differs from the one we offered in the submitted version. Adding the frequency branch as concatenated features is worth +1.53 pp. Adding the frequency MMD on top changes accuracy by -0.57 pp (95% CI $[-1.28, +0.13]$); the interval includes zero, so the appropriate conclusion is that no measurable benefit was detected, not that the term is demonstrably harmful. Adding the cross-view consistency term is worth +2.02 pp, the largest single step, which is attributable, as per Section 4.8, to the coupled consistency-plus-auxiliary mechanism rather than to the consistency loss alone.

The reading we now consider correct is that the frequency branch matters mainly as an enabler: it supplies the second view, without which the consistency constraint cannot be formulated. The two are not separable contributions. For the MMD term, no measurable benefit was detected, and it can be removed; we say so rather than leaving a component in the method because it was in our original design. This is also consistent with AdvSKM's argument [16] that a plain MMD captures only low-order local statistics and is a poor fit to non-stationary time series.

Is it just capacity?

The answer is no. Over a common set of seeds, widening DANN to FTDA's parameter budget shifts accuracy by +0.35 pp and a control matched on parameters and MACs by +0.29 pp—both with intervals that include zero—against FTDA's +2.97 pp on the same runs (Section 4.6). Added capacity alone does not appear to explain the improvement.

The target-only reference.

The target-only reference model reaches 95.71% against FTDA's 95.01%, a gap of 0.70 pp, which is not statistically significant ($p = 0.089$). The oracle is evaluated on the windows it was trained on, so it measures fit rather than generalization; it is not a valid upper bound on target-domain generalization, and we use it only as a loose reference. We therefore do not claim to have reached supervised parity.

Hyper-parameter robustness.

Section 4.13 reports that target accuracy varies by less than ± 1 pp across an order of magnitude in λ_m . Sensitivity to λ_c is larger: setting it to zero costs several points, which is consistent with the consistency term being the principal contributor.

On the $\mathcal{A}$ -distance observation.

Section 4.14 reports that the proxy $\mathcal{A}$ -distance between source and target features on T1 is similar across methods, even though target accuracy differs. We are cautious about what this shows. The measure is a proxy estimated with one particular classifier family (2-fold logistic regression) on a single task and seed, and a stronger domain classifier could yield different values. The defensible statement is, therefore, narrow: under this proxy, no clear difference in domain separability is visible among the methods compared, which suggests that a reduction in $\mathcal{A}$ -distance is not by itself a sufficient explanation for the accuracy differences. It does not establish an alternative explanation, and we treat it as an observation warranting further study rather than as support for the paper's claim.

Limitations.

(i) The two datasets: UCI HAR is small (30 subjects, 6 classes, and smartphone-only), and HHAR covers only device heterogeneity; WISDM, USC-HAD and especially Sleep-EDF, whose signal characteristics differ substantially from activity recognition, would be needed for a general external validity claim. (ii) Shift types evaluated: We evaluated cross-subject shift (UCI HAR) and cross-device shift (HHAR). Environmental shift, sensor placement and cross-dataset transfer are untested, and the HHAR result shows the method's advantage does not automatically carry across shift types. (iii) Simplified re-implementations: The RAINCOAT-style and TF-C-style baselines reproduce the mechanism each paper contributes to our closed-set setting (Table 10), not the full published systems; their numbers should not be read as reproductions of the originals' reported performance. The AdaTime suite (DDC, HoMM, DSAN, and CDAN) is not covered. (iv) FFT magnitude only: Phase information is discarded; phase-aware variants (e.g., complex-valued convolutions) are a natural extension. Section 4.9 also shows that log compression would add a further 0.86 pp, which we did not fold into the main configuration for protocol reasons. (v) No hardware measurement: we report parameter and MAC counts, which do not by themselves establish latency, energy or memory behavior on a real device; on-device claims would require direct measurement. (vi) Ten seeds: Confidence intervals are reported, but the intervals remain wide enough that differences below roughly 1 pp should not be over-interpreted.

6. Conclusions

We present FTDA, a frequency–time joint alignment method for time-series domain adaptation, and evaluate it under a corrected, subject-level protocol with 10 seeds. On UCI

HAR cross-subject transfer, FTDA improves on every single-view baseline on all four tasks under within-task paired tests, while superiority over the two dual-view competitors is not established; a capacity-matched control attributes the gain to the frequency view rather than to parameters.

The ablation locates that gain more precisely than we had originally claimed. The frequency branch is worth +1.53 pp as concatenated features, but +2.02 pp arrives only with the final step, which Section 4.8 shows to be a coupled mechanism: the cross-view consistency term and the auxiliary source supervision operate jointly and must be adopted or dropped together, so we attribute that step to the pair rather than to the consistency loss alone. For the frequency MMD term, no measurable benefit was detected, and it can be removed. The operative mechanism is, therefore, the coupled consistency that the second view enables, not the second view as additional features, and not the alignment losses.

The HHAR results bound the claim. Under cross-device shift, we detect no advantage over time-only adversarial baselines, so the benefit of a spectral view does not carry automatically across shift types. Whether this is because device shift is intrinsically less spectral, or because resampling to a common rate already removed its spectral component, is not something one dataset can settle. Establishing where that boundary lies across a wider range of sensing modalities is the natural next step.

Author Contributions: Conceptualization, K.-S.S. and D.-H.W.; methodology, K.-S.S.; software, K.-S.S.; validation, K.-S.S. and D.-H.W.; formal analysis, K.-S.S. and D.-H.W.; investigation, K.-S.S. and D.-H.W.; resources, K.-S.S.; data curation, K.-S.S.; writing—original draft preparation, D.-H.W.; writing—review and editing, K.-S.S. and D.-H.W.; visualization, K.-S.S.; supervision, K.-S.S.; project administration, K.-S.S.; funding acquisition, D.-H.W. All authors have read and agreed to the published version of the manuscript.

Funding: This paper was supported by Wonkwang University in 2024.

Institutional Review Board Statement: Not applicable. The study uses a publicly available, fully anonymized human-activity dataset (UCI HAR, dataset 240) released by its original collectors with ethical approval at the time of collection.

Informed Consent Statement: Not applicable.

Data Availability Statement: This study analyses two publicly archived datasets and generates no new data. UCI HAR is available from the UCI Machine Learning Repository, DOI <https://doi.org/10.24432/C54S4K>; HHAR (Heterogeneity Activity Recognition) is available from the same repository, DOI <https://doi.org/10.24432/C5689X>. Both were used as distributed, and every preprocessing step needed to reproduce our inputs is specified in the text: the UCI HAR windowing is that of the original release, and the HHAR pipeline (accelerometer channels only, resampling to a common 50 Hz, 128-sample windows at 50% overlap, device model as the domain) is given in Section 4.12. Exact source and target subject identifiers per transfer task, window counts and class distributions are tabulated in Appendix B; the training procedure is given as Algorithm 1 with all loss weights and optimizer settings stated in Sections 3.3 and 3.5; and the auxiliary preprocessing choices are fixed by Equation (1). The full implementation, the configuration files, and the per-run outputs behind every table and figure are archived at Zenodo, DOI <https://doi.org/10.5281/zenodo.21699485>, under the MIT license. The archive does not redistribute either dataset; a download script retrieves both from the UCI repository under their own terms.

Acknowledgments: The authors thank the original collectors of the UCI HAR and HHAR datasets for making them publicly available.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CE	Cross-Entropy
CNN	Convolutional Neural Network
DA	Domain Adaptation
DANN	Domain-Adversarial Neural Network
FFT	Fast Fourier Transform
FTDA	Frequency–Time Domain Alignment (the proposed method)
GRL	Gradient Reversal Layer
HAR	Human Activity Recognition
KL	Kullback–Leibler (divergence)
MAC	Multiply–Accumulate operation
MMD	Maximum Mean Discrepancy
TF-C	Time–Frequency Consistency
UCI	University of California, Irvine
UDA	Unsupervised Domain Adaptation

Appendix A. Step-by-Step Reading of Algorithm 1

This appendix records the implementation-level detail of one FTDA training step. It is included for reproducibility; the design choices that matter for understanding the method are in Section 3.4.

Inputs and outputs. Each step consumes one labeled source mini-batch (X_s, Y_s) and one unlabeled target mini-batch X_t , plus the global step counter t and the total number of steps T used for GRL warm-up. It returns updated parameters for the whole bundle $(g_t, g_f, h, f, D, \phi_t, \phi_f)$ and the scalar loss components for logging. The unlabeled target loader is *cycled*: when its iterator is exhausted within an epoch, it restarts, so every source step is paired with a fresh target mini-batch regardless of the relative sizes of the two domains.

Forward pass (lines 2–3). Both domains pass through both branches. The view-specific features z_t^s, z_t^t and z_f^s, z_f^t are retained for the MMD and consistency terms; the fused features z^s, z^t feed the classifier and the discriminator.

(a) Source classification (line 5). Supervised cross-entropy on the fused source feature. This is the only term that injects label semantics into the encoder; all other terms are either unsupervised or act on the auxiliary heads.

(b) Adversarial alignment (lines 7–9). D is trained to separate source from target, while the GRL negates the gradient reaching the encoder, so the encoder is pushed toward features D cannot separate. $\lambda_{\text{GRL}} = 2/(1 + e^{-10t/T}) - 1$.

(c) Frequency MMD (lines 10–11). Multi-kernel Gaussian MMD² between z_f^s and z_f^t , with bandwidths set by the median heuristic scaled by $\{1, 2, 4, 8\}$.

(d) Cross-view consistency (lines 13–14). Symmetric KL between $\text{softmax}(\phi_t(z_t^t))$ and $\text{softmax}(\phi_f(z_f^t))$, evaluated on unlabeled target windows only.

(e) Auxiliary source supervision (line 16). Cross-entropy on $\phi_t(z_t^s)$ and $\phi_f(z_f^s)$, weighted by λ_x . See Section 4.8 for why this term is required rather than optional.

Composite loss and update (lines 18–19). The five components are summed with the weights of Equation (2) and optimized in a single backward pass, so the objectives trade off through shared parameters rather than through alternating updates; Figure 6 shows the resulting dynamics. Model selection uses source-validation accuracy only, so target labels never enter the procedure.

Appendix B. Exact Experimental Protocol

This appendix records the partitions and counts needed to reproduce the experiments without access to our code: Table A1 gives the exact subject partitions and Table A2 the class distributions.

Table A1. UCI HAR cross-subject transfer tasks: exact subject identifiers and window counts. Subjects are disjoint between source and target in every task. Windows are the 128-sample, 50%-overlap segments of the original release.

Task	Source Subjects	n_{src}	Target Subjects	n_{tgt}
T1	1, 3, 5, 6, 7, 8, 11, 14, 15, 16, 17	3605	2, 4, 9, 10, 12, 13	1848
T2	19, 21, 22, 23, 25, 26, 27, 28, 29, 30	3747	18, 20, 24	1099
T3	1–15	4719	16–30	5580
T4	16–30	5580	1–15	4719

Within the source domain, 20% of the subjects (not windows) are held out for validation; see Section 3.8. The leave-one-subject-out study of Section 4.11 uses subjects 1–10, each in turn as the sole target with the other nine as source.

Table A2. Class distributions. UCI HAR totals 10,299 windows over 30 subjects; HHAR totals 76,647 windows over 9 subjects and 4 device models.

UCI HAR			HHAR (Phone Accelerometer)		
Class	n	%	Class/Device	n	%
WALKING	1722	16.7	bike	12,630	16.5
WALKING_UPSTAIRS	1544	15.0	sit	13,867	18.1
WALKING_DOWNSTAIRS	1406	13.7	stand	13,098	17.1
SITTING	1777	17.3	walk	14,655	19.1
STANDING	1906	18.5	stairsup	11,691	15.3
LAYING	1944	18.9	stairsdown	10,706	14.0
			nexus4	19,647	25.6
			s3	20,915	27.3
			s3mini	11,425	14.9
			samsunggold	24,660	32.2

Neither dataset is class-balanced, but neither is severely skewed: the rarest class holds 13.7% of UCI HAR and 14.0% of HHAR. We therefore report accuracy alongside macro-F1 rather than applying any class re-weighting.

References

1. Ganin, Y.; Lempitsky, V. Unsupervised Domain Adaptation by Backpropagation. In Proceedings of the 32nd International Conference on Machine Learning (ICML), Lille, France, 6–11 July 2015; Volume 37, pp. 1180–1189. Available online: <https://proceedings.mlr.press/v37/ganin15.html> (accessed on 2 August 2026).
2. Ben-David, S.; Blitzer, J.; Crammer, K.; Kulesza, A.; Pereira, F.; Vaughan, J.W. A Theory of Learning from Different Domains. *Mach. Learn.* **2010**, *79*, 151–175. [CrossRef]
3. Long, M.; Cao, Y.; Wang, J.; Jordan, M.I. Learning Transferable Features with Deep Adaptation Networks. In Proceedings of the 32nd International Conference on Machine Learning (ICML), Lille, France, 6–11 July 2015; Volume 37, pp. 97–105. Available online: <https://proceedings.mlr.press/v37/long15.html> (accessed on 2 August 2026).
4. He, H.; Queen, O.; Koker, T.; Cuevas, C.; Tsiligkaridis, T.; Zitnik, M. Domain Adaptation for Time Series Under Feature and Label Shifts. In Proceedings of the 40th International Conference on Machine Learning (ICML), Honolulu, HI, USA, 23–29 July 2023; Volume 202. Available online: <https://proceedings.mlr.press/v202/he23b.html> (accessed on 2 August 2026).

5. Zhang, X.; Zhao, Z.; Tsiligkaridis, T.; Zitnik, M. Self-Supervised Contrastive Pre-Training for Time Series via Time-Frequency Consistency. In Proceedings of the Advances in Neural Information Processing Systems 35 (NeurIPS), New Orleans, LA, USA, 28 November–9 December 2022. Available online: <https://openreview.net/forum?id=OJ4mMfGKLN4> (accessed on 2 August 2026).
6. Anguita, D.; Ghio, A.; Oneto, L.; Parra, X.; Reyes-Ortiz, J.L. A Public Domain Dataset for Human Activity Recognition Using Smartphones. In Proceedings of the European Symposium on Artificial Neural Networks (ESANN), Bruges, Belgium, 24–26 April 2013; UCI Machine Learning Repository, dataset 240. [CrossRef]
7. Wilson, G.; Doppa, J.R.; Cook, D.J. Multi-Source Deep Domain Adaptation with Weak Supervision for Time-Series Sensor Data. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual*, 6–10 July 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 1768–1778. [CrossRef]
8. Tzeng, E.; Hoffman, J.; Saenko, K.; Darrell, T. Adversarial Discriminative Domain Adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA*, 21–26 July 2017; IEEE: New York, NY, USA, 2017; pp. 7167–7176. [CrossRef]
9. Long, M.; Cao, Z.; Wang, J.; Jordan, M.I. Conditional Adversarial Domain Adaptation. In Proceedings of the Advances in Neural Information Processing Systems 31 (NeurIPS), Montréal, QC, Canada, 3–8 December 2018; pp. 1647–1657. Available online: https://papers.nips.cc/paper_files/paper/2018/hash/ab88b15733f543179858600245108dd8-Abstract.html (accessed on 2 August 2026).
10. Purushotham, S.; Carvalho, W.; Nilanon, T.; Liu, Y. Variational Recurrent Adversarial Deep Domain Adaptation. In Proceedings of the International Conference on Learning Representations (ICLR), Toulon, France, 24–26 April 2017. Available online: <https://openreview.net/forum?id=rk9eAFcxg> (accessed on 2 August 2026).
11. Gretton, A.; Borgwardt, K.M.; Rasch, M.J.; Schölkopf, B.; Smola, A. A Kernel Two-Sample Test. *J. Mach. Learn. Res.* **2012**, *13*, 723–773. Available online: <https://jmlr.org/papers/v13/gretton12a.html> (accessed on 2 August 2026).
12. Sun, B.; Saenko, K. Deep CORAL: Correlation Alignment for Deep Domain Adaptation. In *Proceedings of the Computer Vision–ECCV 2016 Workshops, Amsterdam, The Netherlands, 8–16 October 2016*; Springer: Cham, Switzerland, 2016; pp. 443–450. [CrossRef]
13. Shen, J.; Qu, Y.; Zhang, W.; Yu, Y. Wasserstein Distance Guided Representation Learning for Domain Adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA*, 2–7 February 2018; AAAI Press: Washington, DC, USA, 2022; Volume 32. [CrossRef]
14. Courty, N.; Flamary, R.; Tuia, D.; Rakotomamonjy, A. Optimal Transport for Domain Adaptation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1853–1865. [CrossRef] [PubMed]
15. Cuturi, M. Sinkhorn Distances: Lightspeed Computation of Optimal Transport. In Proceedings of the Advances in Neural Information Processing Systems 26 (NeurIPS), Lake Tahoe, NV, USA, 5–10 December 2013; pp. 2292–2300. Available online: <https://proceedings.neurips.cc/paper/2013/hash/af21d0c97db2e27e13572cbf59eb343d-Abstract.html> (accessed on 2 August 2026).
16. Liu, Q.; Xue, H. Adversarial Spectral Kernel Matching for Unsupervised Time Series Domain Adaptation. In Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI), Montreal, QC, Canada, 19–27 August 2021; pp. 2744–2750. [CrossRef] [PubMed]
17. Yue, Z.; Wang, Y.; Duan, J.; Yang, T.; Huang, C.; Tong, Y.; Xu, B. TS2Vec: Towards Universal Representation of Time Series. In *Proceedings of the AAAI Conference on Artificial Intelligence, Virtual*, 22 February–1 March 2022; AAAI Press: Washington, DC, USA, 2022; Volume 36, pp. 8980–8987. [CrossRef]
18. Eldele, E.; Ragab, M.; Chen, Z.; Wu, M.; Kwok, C.K.; Li, X.; Guan, C. Time-Series Representation Learning via Temporal and Contextual Contrasting. In Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI), Montreal, QC, Canada, 19–27 August 2021; pp. 2352–2359. [CrossRef] [PubMed]
19. Ozyurt, Y.; Feuerriegel, S.; Zhang, C. Contrastive Learning for Unsupervised Domain Adaptation of Time Series. In Proceedings of the International Conference on Learning Representations (ICLR), Kigali, Rwanda, 1–5 May 2023. Available online: <https://openreview.net/forum?id=xPkJYRsQGM> (accessed on 2 August 2026).
20. Tarvainen, A.; Valpola, H. Mean Teachers Are Better Role Models: Weight-Averaged Consistency Targets Improve Semi-Supervised Deep Learning Results. In Proceedings of the Advances in Neural Information Processing Systems 30 (NeurIPS), Long Beach, CA, USA, 4–9 December 2017; pp. 1195–1204. Available online: https://papers.nips.cc/paper_files/paper/2017/hash/68053af2923e00204c3ca7c6a3150cf7-Abstract.html (accessed on 2 August 2026).
21. Wang, D.; Shelhamer, E.; Liu, S.; Olshausen, B.; Darrell, T. Tent: Fully Test-Time Adaptation by Entropy Minimization. In Proceedings of the International Conference on Learning Representations (ICLR), Virtual, 3–7 May 2021. Available online: <https://openreview.net/forum?id=uXl3bZLkr3c> (accessed on 2 August 2026).

22. Ragab, M.; Eldele, E.; Tan, W.L.; Foo, C.-S.; Chen, Z.; Wu, M.; Kwok, C.-K.; Li, X. ADATIME: A Benchmarking Suite for Domain Adaptation on Time Series Data. *ACM Trans. Knowl. Discov. Data* **2023**, *17*, 1–18. [[CrossRef](#)]
23. Stisen, A.; Blunck, H.; Bhattacharya, S.; Prentow, T.S.; Kjærgaard, M.B.; Dey, A.; Sonne, T.; Jensen, M.M. Smart Devices Are Different: Assessing and Mitigating Mobile Sensing Heterogeneities for Activity Recognition. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems (SenSys), Seoul, Republic of Korea, 1–4 November 2015*; Association for Computing Machinery: New York, NY, USA, 2015; pp. 127–140. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.