

BayaHAR: Lightweight Bayesian Few-Shot User Adaptation for On-Device Personalized Human Activity Recognition

Maximilian Burzer
maximilian.burzer@kit.edu
Karlsruhe Institute of Technology
Karlsruhe, Germany

Michael Beigl
michael.beigl@kit.edu
Karlsruhe Institute of Technology
Karlsruhe, Germany

Till Riedel
till.riedel@kit.edu
Karlsruhe Institute of Technology
Karlsruhe, Germany

Tobias Röddiger
tobias.roeddiger@ipai-foundation.ai
IPAI Foundation gGmbH
Heilbronn, Germany

Abstract

Sensor-based Human Activity Recognition (HAR) models often degrade on unseen users due to domain shifts caused by individual movement patterns and sensor placement. Practical wearable HAR systems therefore require personalization methods that are lightweight, applicable across diverse calibration scenarios, and robust under limited calibration data. We present BayaHAR, a gradient-free framework that repurposes pretrained HAR classifiers as Prototypical Networks using prior prototypes that preserve zero-shot performance while regularizing adaptation. For labeled calibration data, we introduce closed-form Bayesian prototype estimation and extend the same principle to weakly labeled data, requiring only knowledge of which activities were performed. With only 3 seconds of calibration (one shot) per activity, supervised adaptation improves test macro-F1 on unseen users by +2.76 to +33.44 percentage points across four datasets, while weakly supervised adaptation improves by +0.56 to +32.13 points. Since adaptation requires only closed-form prototype updates, the framework enables efficient and robust on-device personalization of preexisting HAR classifiers.

CCS Concepts

• **Human-centered computing** → Ubiquitous and mobile computing; • **Computing methodologies** → Learning paradigms.

Keywords

Human Activity Recognition; Domain Adaptation; Few-Shot Learning; Prototypical Networks; Bayesian Inference

ACM Reference Format:

Maximilian Burzer, Till Riedel, Michael Beigl, and Tobias Röddiger. 2026. BayaHAR: Lightweight Bayesian Few-Shot User Adaptation for On-Device Personalized Human Activity Recognition. In *Proceedings of the 2026 ACM International Symposium on Wearable Computers (ISWC '26)*, October 11–15, 2026, Shanghai, China. ACM, New York, NY, USA, 7 pages. <https://doi.org/10.1145/3830727.3834833>



This work is licensed under a Creative Commons Attribution 4.0 International License. *ISWC '26, Shanghai, China*

© 2026 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-2872-3/2026/10

<https://doi.org/10.1145/3830727.3834833>

1 Introduction

Sensor-based Human Activity Recognition (HAR) models often degrade when deployed on unseen users due to domain shifts caused by inter-subject variability, including physiological differences in movement patterns and variations in sensor placement [9, 12]. Practical HAR systems therefore require adaptation mechanisms that remain effective across different calibration scenarios. At the same time, personalization should be lightweight enough for resource-constrained wearable devices, where latency, memory, and battery consumption limit the use of gradients [20].

To address these requirements, we build on Prototypical Networks [31], an efficient framework for few-shot classification. However, conventional Prototypical Networks assume that a labeled support set is available during inference and therefore lack a natural zero-shot fallback when no calibration data is provided. Moreover, standard prototype estimation relies on empirical support means, which can become unstable in the ultra-low-shot regime due to sample noise and high variance [40]. These issues are particularly problematic for HAR, where calibration should be as short as possible and weakly labeled recordings, with only knowledge on which activities were performed, are often more realistic [30].

We therefore introduce BayaHAR (Bayesian Adaptive HAR), a gradient-free framework that repurposes pretrained HAR classifiers as Prototypical Networks with source-domain prior prototypes. These priors preserve the original zero-shot behavior without calibration data and serve as regularizers when labeled or weakly labeled support samples are introduced. For labeled data, we update prototypes through closed-form Bayesian Maximum A Posteriori (MAP) estimation. For weakly labeled data, we combine this principle with an Expectation-Maximization (EM) update over latent activity class assignments. In both cases, the prior enables robust adaptation in low-support regimes while retaining a low computational overhead. We publicly release our code and experimental setup¹. Our key contributions are summarized as follows:

(1) *Repurposing via Prior Prototypes*: We introduce prior prototypes, which allow existing pretrained neural HAR classifiers to be repurposed as Prototypical Networks, while retaining their standard classifier behavior. This preserves zero-shot performance with negligible computational overhead and provides a unified starting point for few-shot adaptation.

¹<https://github.com/maxbrzr/baya-har>

(2) *MAP Prototype Estimation*: We introduce closed-form MAP prototype estimation for labeled calibration data, regularizing noisy support estimates. With one shot (3 seconds) per activity, this improves macro-F1 on four datasets, with gains from +2.76 to +33.44 pp. With 16 shots (48 seconds), gains range from +6.8 to +33.7 pp.

(3) *MAP-EM Prototype Estimation*: We extend prototype estimation to weakly labeled calibration data using single-step MAP-EM, requiring only knowledge about which activity classes are present. With one shot (3 seconds) per activity, MAP-EM improves macro-F1 over the zero-shot prior, with gains from +0.56 to +32.13 pp, while avoiding degradation observed in baselines. With 16 shots (48 seconds), gains range from +0.9 to +32.5 pp.

2 Related Work

In sensor-based HAR, deploying models on unseen users introduces significant domain shifts [9, 12]. These shifts primarily emerge from variations in sensor placement, which induce covariate shifts, and physiological differences in how subjects perform movements, which induce class-conditional shifts. While domain adaptation has been extensively researched to address these overlapping challenges, practical deployment on wearable devices introduces strict constraints that limit the applicability of many standard techniques.

Due to the strict memory, computational, and energy constraints of wearable devices, on-device user adaptation must be *source-free*: relying solely on pretrained model parameters without requiring access to the original training data. This requirement disqualifies conventional domain adaptation methods that rely on source data, such as Domain-Adversarial Neural Networks (DANN) [11], and Deep CORAL [34]. Furthermore, on-device adaptation necessitates *gradient-free* approaches. Backpropagation is computationally expensive and popular mobile machine learning frameworks like ExecuTorch [26] and TensorFlow Lite [35] are designed primarily for inference and lack native backpropagation support. This constraint eliminates parameter-efficient fine-tuning approaches like LoRA [13], optimization-based meta-learning like MAML [10] and prominent gradient-based test-time adaptation (TTA) methods such as SHOT [18] and Tent [37]. While recent gradient-free TTA methods like OFFTA [38] and NEO [23] bypass backpropagation, they perform continuous online adaptation, which can be problematic on uncontrolled sensor streams, where updates may accumulate errors, reinforce noisy pseudo-labels, and cause catastrophic forgetting [25, 39]. These risks are particularly relevant in HAR, where incoming windows are often temporally correlated and low-diversity, so adaptation may overfit to recent activities or contexts rather than remain stable over longer sessions [27]. Our work therefore targets a dedicated *offline* calibration phase, using a limited set of user-specific calibration data, after which the adapted model is fixed to ensure stable personalization.

To enable *source-free*, *gradient-free*, and *offline* adaptation, few-shot approaches rooted in Prototypical Networks [31] offer a promising direction. While originally supervised, unsupervised extensions such as PDA [6] have also been developed. Within HAR, prototype-based methods have predominantly been explored in Federated Learning contexts [8, 28, 32] or applied to continual learning scenarios for adapting to novel activity classes [1, 14, 15], which is beyond our scope. Recent works [17, 19] have applied prototype-based adaptation to address domain shifts in HAR, however they are neither

source- nor gradient-free. Moreover, all mentioned prototype-based methods fundamentally require support sets and cannot fallback to standard HAR classifiers when calibration data is unavailable.

Related few-shot paradigms such as self-supervised HAR foundation models also leverage reusable embedding spaces for few-shot transfer, but focus on broader cross-dataset, cross-modal, cross-placement, or language-grounded generalization [2]. They are orthogonal to our work, as our adaptation strategy could be applied on top of such pretrained embeddings.

To the best of our knowledge, no existing approach fulfills all the mentioned requirements for practical offline user adaptation in HAR on wearable devices: being source-free, gradient-free, and computationally efficient, while flexibly and robustly handling zero-shot, supervised, and weakly supervised calibration scenarios. Building upon Prototypical Networks [31], our work bridges this gap.

3 Prototypical Networks

Prototypical Networks [31] (ProtoNets) constitute a few-shot and metric-based meta-learning framework that maps raw data into a learned d -dimensional continuous embedding space using an embedding network f_θ . In this space, embeddings of samples belonging to the same class k , cluster around a single prototype representation c_k . Given a support set containing support examples S_k for every class k , the prototypes are computed as the embedding mean:

$$c_k = \frac{1}{|S_k|} \sum_{x \in S_k} f_\theta(x). \quad (1)$$

To classify an unseen query sample x_q , its embedding $f_\theta(x_q)$ is compared to each prototype c_k , as illustrated in Fig. 1. For this purpose, the standard formulation uses the squared Euclidean distance:

$$d(f_\theta(x_q), c_k) = \|f_\theta(x_q) - c_k\|^2. \quad (2)$$

The probability $p(y = k | x)$ of query sample x belonging to class k is then computed using a softmax over the negative distances to the prototypes. Consequently, the prototype with the smallest distance corresponds to the predicted class.

The embedding network f_θ is typically trained using episodic training [31], a common strategy in meta-learning. During each episode, support and query sets are sampled in pairs and passed through the model, mimicking the few-shot setting and encouraging the model to learn how to generalize from limited labeled examples rather than memorizing large datasets.

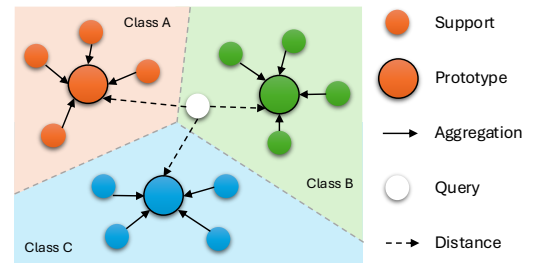


Figure 1: During inference, the query sample is compared to three prototypes computed as the means of their respective support examples in embedding space.

4 Methodology

In the following, we describe repurposing of ProtoNets with prior prototypes for when no calibration data is available (Sec. 4.1). We then show how these priors can be updated with labeled (Sec. 4.2) or weakly labeled (Sec. 4.3) support examples.

4.1 Prior Prototypes

While ProtoNets [31] were originally designed for few-shot learning to accommodate entirely novel classes, our work leverages their architecture for user adaptation instead. By restricting our scope to a fixed set of predefined activity classes, we eliminate the strict requirement for a target-domain support set during inference, unlocking significantly greater flexibility for real-world deployments.

To enable classification when user-specific calibration data is unavailable, we propose defining a set of *prior prototypes*. Instead of relying on a support set S_k , we compute a global prototype per class k as the mean of all embedded training data points D_k , replacing the support examples in Eq. 1. Therefore, this repurposing step itself is not *source-free*, but can be performed offline before deployment. The resulting prior prototypes are stored on-device and act as a highly efficient zero-shot fallback. During inference, evaluating a query embedding against these prototypes incurs negligible computational overhead compared to the final linear classification layer in standard HAR models, as it closely mirrors the matrix multiplication. As demonstrated in our experiments, this imposes essentially no performance penalty compared to the original model, while accommodating future adaptation if calibration data is introduced.

Crucially, this formulation allows existing pretrained deep learning classifiers, such as TinierHAR [3], to be seamlessly repurposed as ProtoNets. By simply removing the final linear classification layer, the pretrained backbone functions directly as the embedding network within the ProtoNet framework. Although standard ProtoNets are typically optimized using specialized episodic meta-training, prior work [16] argues that standard training with cross-entropy loss is sufficient to yield a well-clustered embedding space. Our experiments specifically validate this finding for sensor-based HAR. This flexibility allows developers to select from a wide variety of performant, state-of-the-art HAR models and repurpose them without the need for complex retraining from scratch.

Beyond acting as a zero-shot fallback, these prior prototypes preserve source-domain information that standard ProtoNets discard when adapting only from support examples. Once computed and stored on-device, they can therefore additionally serve as regularizers for low-data and noisy calibration.

4.2 MAP Prototype Estimation

Standard ProtoNets estimate each class prototype as the empirical mean of its support embeddings (see Sec. 3), which can be interpreted as a Maximum Likelihood Estimate (MLE) fitted only to the support set. To incorporate source-domain information from prior prototypes, we instead formulate prototype estimation as a Bayesian inference problem [24] and compute each prototype as Maximum A Posteriori (MAP) estimate:

$$c_k^{\text{MAP}} = \arg \max_{c_k} p(c_k | S_k) = \arg \max_{c_k} [p(S_k | c_k) p(c_k)]. \quad (3)$$

We model the source-domain prior distribution $p(c_k)$ and support likelihood $p(S_k | c_k)$ as Gaussians:

$$p(c_k) = \mathcal{N}(c_k | \mu_{D,k}, \sigma_{D,k}^2), \quad p(S_k | c_k) \propto \mathcal{N}\left(\mu_{S,k} | c_k, \frac{\sigma_{S,k}^2}{N}\right). \quad (4)$$

Here, $N = |S_k|$, while $\mu_{D,k}$ and $\sigma_{D,k}^2$ denote the empirical mean and variance estimated from the embeddings of the training examples D_k , while $\mu_{S,k}$ and $\sigma_{S,k}^2$ are derived from the support examples S_k . The variance $\sigma_{S,k}^2/N$ represents the uncertainty of the empirical support mean, so the likelihood becomes more precise as the support size increases. To maintain computational efficiency, we constrain all covariance matrices to be diagonal and represent the corresponding variances as vectors. For $N = 1$, the empirical support variance is undefined, so we set $\sigma_{S,k}^2 = \sigma_{D,k}^2$.

Because the Gaussian prior and likelihood are conjugate, the posterior $p(c_k | S_k)$ is also Gaussian [24]. Its mode coincides with its mean, such that $c_k^{\text{MAP}} = \mu_{\text{post},k}$, yielding the following closed-form expressions:

$$\frac{1}{\sigma_{\text{post},k}^2} = \frac{1}{\sigma_{D,k}^2} + \frac{N}{\sigma_{S,k}^2}, \quad (5)$$

$$c_k^{\text{MAP}} = \mu_{\text{post},k} = \left(\frac{\sigma_{\text{post},k}^2}{\sigma_{D,k}^2}\right) \mu_{D,k} + \left(\frac{N \sigma_{\text{post},k}^2}{\sigma_{S,k}^2}\right) \mu_{S,k} \quad (6)$$

Intuitively, Eq. 6 can be viewed as a precision-weighted combination of the source-domain prior mean and the target-user support mean. A simpler alternative would be a fixed-weight interpolation:

$$c_k^{\text{fix}} = (1 - \alpha) \mu_{D,k} + \alpha \mu_{S,k} \quad \text{with } \alpha \in [0, 1]. \quad (7)$$

However, this alternative assigns the same trust to the support set regardless of user, activity class, embedding dimension, support size, or calibration noise. It may therefore over-adapt to noisy or unrepresentative support examples, moving prototypes too far away from the source-domain prior prototypes, or under-adapt when support examples are consistent and informative.

In contrast, MAP prototype estimation derives its weights from the relative precisions of the prior and support statistics. This weighting, and thus the degree of adaptation, is class-, dimension-, and data-dependent: when support evidence exhibits statistical noise, the posterior reflects higher epistemic uncertainty about the target prototype and regularizes the estimate toward the prior, improving robustness against outliers, while precise support evidence receives greater influence. Furthermore, this avoids introducing the additional hyperparameter α , which would be difficult to tune reliably for unseen users. Since the update is analytical, gradient-free, and efficient to compute, it adds only a low computational overhead on resource-constrained wearable devices (see Sec. 5.1). The complete step-by-step procedure is outlined in Algo. 1.

Algorithm 1 MAP Prototype Estimation

- 1: Compute support mean $\mu_{S,k}$ and variance $\sigma_{S,k}^2$
 - 2: Compute posterior variance $\sigma_{\text{post},k}^2$ using Eq. 5
 - 3: Compute posterior mean $\mu_{\text{post},k}$ using Eq. 6
 - 4: **return** $c_k^{\text{MAP}} = \mu_{\text{post},k}$
-

4.3 MAP-EM Prototype Estimation

MAP prototype estimation enables robust few-shot adaptation when a labeled support set is available. However, while recording calibration data is feasible in many HAR settings, labeling each sample is often impractical [30]. We therefore extend MAP prototype estimation to weakly labeled support sets, where only information on which activities were performed is provided. In this setting, we frame the unknown labels $Z = \{z_i\}$ of support embeddings $S = \{s_i\}$ as latent variables. This gives a Gaussian Mixture Model in the embedding space, where each present activity class corresponds to one Gaussian component, and $z_i \in \{1, \dots, K\}$ indicates which component generated s_i . We use uniform mixture weights $\pi_k = 1/K$ and an efficient fixed isotropic covariance $\Sigma_k = \sigma_{\text{EM}}^2 I$. Since these quantities are fixed, the only parameters optimized during adaptation are the component means μ_k , which correspond to updated prototypes $C = \{c_k\}$. To fit the prototypes to the weakly labeled samples, we define the MAP objective:

$$C^{\text{MAP}} = \arg \max_C \left[\underbrace{\log p(S|C)}_{\sum_{i=1}^{N_S} \log \sum_{k=1}^K \pi_k \mathcal{N}(s_i | c_k, \sigma_{\text{EM}}^2 I)} + \underbrace{\log p(C)}_{\sum_{k=1}^K \log \mathcal{N}(c_k | \mu_{D,k}, \sigma_{D,k}^2)} \right]. \quad (8)$$

This objective again incorporates source-domain information using the same Gaussian prototype priors $p(c_k)$ (see Eq. 4). Direct optimization is difficult because the class assignments z_i are unobserved, so each support likelihood contains a sum over all possible assignments inside the logarithm. This couples the prototype updates and prevents a simple closed-form solution.

We therefore utilize Expectation-Maximization (EM) [4], an iterative optimization algorithm used to estimate model parameters in the presence of latent variables given observed data. The E-step computes the support posterior probabilities, or soft responsibilities, for each support embedding:

$$r_{ik}^{(t)} = p(z_i = k | s_i, C^{(t)}) = \frac{\pi_k \mathcal{N}(s_i | c_k^{(t)}, \sigma_{\text{EM}}^2 I)}{\sum_{j=1}^K \pi_j \mathcal{N}(s_i | c_j^{(t)}, \sigma_{\text{EM}}^2 I)}. \quad (9)$$

Based on these responsibilities, the M-step updates the prototypes using the support evidence together with the prototype prior. First, we compute the soft count and responsibility-weighted support mean for each class:

$$N_k^{(t)} = \sum_{i=1}^{N_S} r_{ik}^{(t)} \quad \text{and} \quad \mu_{S,k}^{(t)} = \frac{1}{N_k^{(t)}} \sum_{i=1}^{N_S} r_{ik}^{(t)} s_i. \quad (10)$$

The prototype update then is the MAP estimate of the component mean c_k under the prototype prior $p(c_k)$ and the responsibility-weighted support evidence. It follows the same closed-form equations as for MAP prototype estimation (see Eq. 5, Eq. 6), but replaces the labeled support statistics with their soft EM counterparts:

$$\frac{1}{\sigma_{\text{post},k}^{2(t+1)}} = \frac{1}{\sigma_{D,k}^2} + \frac{N_k^{(t)}}{\sigma_{\text{EM}}^2}, \quad (11)$$

$$c_k^{(t+1)} = \mu_{\text{post},k}^{(t+1)} = \left(\frac{\sigma_{\text{post},k}^{2(t+1)}}{\sigma_{D,k}^2} \right) \mu_{D,k} + \left(\frac{N_k^{(t)} \sigma_{\text{post},k}^{2(t+1)}}{\sigma_{\text{EM}}^2} \right) \mu_{S,k}^{(t)}. \quad (12)$$

The soft count $N_k^{(t)}$ acts as the effective number of support examples assigned to class k . Confident responsibilities increase the influence of the support evidence and move the prototype toward the target-subject support distribution, while uncertain assignments or small soft counts leave the prior prototype mean $\mu_{D,k}$ with greater influence. After a fixed number of EM iterations, the final prototype estimates are $C^{(T)} = \{c_k^{(T)}\}$.

Before applying MAP-EM prototype estimation and final query classification, we center the embedding space to reduce subject-specific distribution shifts. For the train set, this can be viewed as centering each prototype with the global train embedding mean $\bar{d}$, and support and query embeddings with the support mean $\bar{s}$:

$$\tilde{\mu}_{D,k} = \mu_{D,k} - \bar{d}, \quad \tilde{s}_i = s_i - \bar{s}, \quad \tilde{q}_i = q_i - \bar{s}. \quad (13)$$

This removes global subject-specific offsets that affect all activities in a similar direction, allowing MAP-EM to focus on class-specific changes in the target subject rather than compensating for a global shift that can be easily corrected by centering. Overall, the complete step-by-step procedure is outlined in Algo. 2.

Algorithm 2 MAP-EM Prototype Estimation

- 1: Center support embeddings S using Eq. 13
 - 2: Initialize prototypes $c_k^{(0)} = \tilde{\mu}_{D,k}$
 - 3: **for** $t = 0 \dots T - 1$ **do**
 - 4: Compute responsibilities $r_{ik}^{(t)}$ using Eq. 9
 - 5: Compute soft count $N_k^{(t)}$, support mean $\mu_{S,k}^{(t)}$ using Eq. 10
 - 6: Compute posterior variance $\sigma_{\text{post},k}^{2(t+1)}$ using Eq. 11
 - 7: Compute posterior mean $\mu_{\text{post},k}^{(t+1)} = c_k^{(t+1)}$ using Eq. 12
 - 8: **end for**
 - 9: **return** $C^{(T)} = \{c_k^{(T)}\}$
-

5 Evaluation

We evaluate on four HAR benchmark datasets (see Table 1) using the WHAR Datasets library [7] with 3-second windows and leave-one-subject-out (LOSO) cross-validation. For each fold, the held-out subject is used exclusively for testing, while the remaining subjects are split into 80% training and 20% validation data. We use the efficient TinierHAR [3] as base classifier and train each model using AdamW [22] with cross-entropy loss for up to 100 epochs, batch size 64, learning rate 10^{-4} , 50% window overlap, and early stopping

Table 1: HAR datasets used for benchmarking.

Dataset	Subjects	Activities	Channels	Sensors
HHAR [33]	9	6	6	Acc, Gyr
WEAR [5]	24	18	12	Acc
HARTH [21]	22	12	6	Acc
HAPT [29]	30	12	6	Acc, Gyr

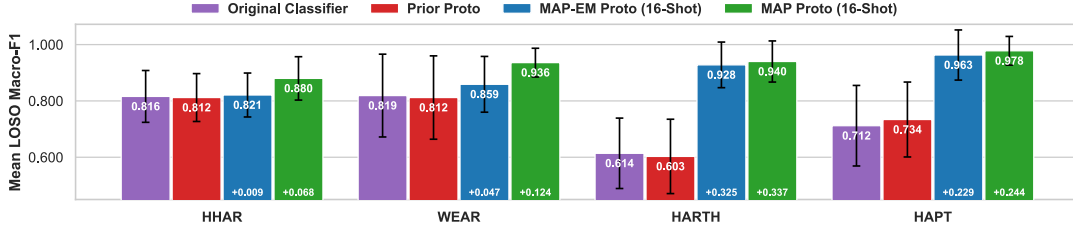


Figure 2: Mean LOSO macro-F1 across four HAR datasets for the original classifier, its repurposing with prior prototypes, and our proposed weakly supervised (MAP-EM Proto) and supervised (MAP Proto) adaptation methods on 16 shots per class.

on validation macro-F1 with patience 10 for model selection. We report mean LOSO macro-F1 and standard deviation. For few-shot evaluation, we disable window overlap and average results over 100 episodes for 1 to 16 shots per class. Support and query sets are strictly disjoint, and all samples not selected for support are used as queries. For MAP-EM, we use one iteration with $\sigma_{EM}^2 = 0.5$.

As shown in Fig. 2, prior prototypes successfully repurpose the pretrained classifier as a ProtoNet while preserving zero-shot performance. With 16 labeled shots (48 seconds of calibration) per activity, MAP prototype estimation yields substantial test macro-F1 gains across all datasets, ranging from +6.8 pp on HHAR to +33.7 pp on HARTH. MAP-EM achieves comparable gains, ranging from +0.9 pp on HHAR to +32.5 pp on HARTH. Fig. 3 illustrates this behavior via T-SNE [36]: the support embeddings are initially shifted away from the prior prototypes due to domain shift, while a single MAP-EM iteration moves the prototypes toward the target-subject clusters and repairs the decision boundaries. Both methods also visibly reduce LOSO variance compared to the zero-shot baseline.

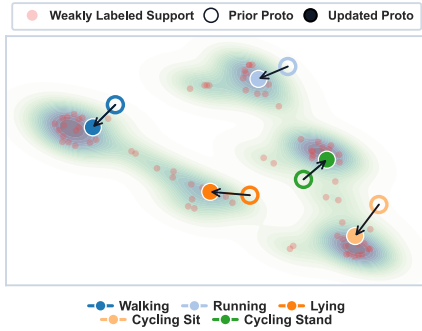


Figure 3: T-SNE [36] visualization of one MAP-EM update on HARTH with 16 weakly labeled support examples per class.

Fig. 4 illustrates how macro-F1 performance scales with number of support examples per class. We compare against several baselines, all based on the same pretrained classifier backbones. For the supervised setting, we compare against standard ProtoNets [31] and a logistic regression probe fitted to the calibration data. The latter is gradient-based and therefore unsuitable for on-device adaptation, but serves as an approximate performance ceiling. For the weakly supervised setting, we compare against weakly supervised variants of PDA [6], which builds prototypes from pseudo-labeled support data, and OFTTA [38], a HAR-specific TTA method that we adapted to the offline setting by freezing updates after calibration.

Looking closely at the curves in Fig. 4, adaptation performance largely converges around 16 support shots per class across all methods. At these higher support sizes, our proposed approaches match the peak performance of the established baselines. The logistic probe establishes the overall performance ceiling on HHAR, WEAR, and HAPT by leveraging gradients. However, stark differences emerge in the extreme low-shot regime of just one shot per class (only 3 seconds). Table 2 reports the corresponding macro-F1 gains over the prior-prototype zero-shot baseline. Both proposed methods improve from the first shot on all datasets: MAP gains between +2.76 pp on HHAR and +33.44 pp on HARTH, while MAP-EM remains consistently positive with gains from +0.56 pp to +32.13 pp. In contrast, the supervised baselines are less reliable, with ProtoNets and the logistic probe falling below the zero-shot baseline on HHAR by -2.65 pp and -3.27 pp, respectively. This vulnerability is even more pronounced in the weakly supervised setting: OFTTA and PDA collapse below the baseline on both HHAR and WEAR, with PDA dropping by -12.81 pp on HHAR and -6.48 pp on WEAR. Even where these baselines still improve, MAP-EM achieves the strongest one-shot performance, highlighting its ability to avoid early-stage degradation in the ultra-low-shot regime.

Table 2: Macro-F1 gains over the prior-prototype zero-shot baseline at one shot (3 seconds of calibration) per activity.

Dataset	Supervised			Weakly Supervised		
	MAP	ProtoNet	Log. Probe	MAP-EM	OFTTA	PDA
HAPT	+23.42	+22.63	+22.74	+22.67	+21.55	+20.49
HARTH	+33.44	+30.73	+30.88	+32.13	+28.91	+26.84
HHAR	+2.76	-2.65	-3.27	+0.56	-3.57	-12.81
WEAR	+8.12	+5.37	+5.26	+3.29	-0.76	-6.48

5.1 Computational Overhead

We quantify only the prototype estimation cost after embedding extraction. MAP scales roughly as $O((N + K)d)$, and MAP-EM as $O(TNKd)$, with K adapted classes, N support embeddings of dimension d , and T EM steps. We measure the concrete values shown in Table 3 with an operation-level counter matching our implementation. Across 1 to 16 shots, MAP requires only 0.3–3.9% of a single TinierHAR backbone forward pass. MAP-EM is more expensive, but also remains below one pass, requiring 1.4–82.4%. This low cost is enabled by diagonal covariance in MAP and isotropic covariance in MAP-EM: although full covariances could model correlations

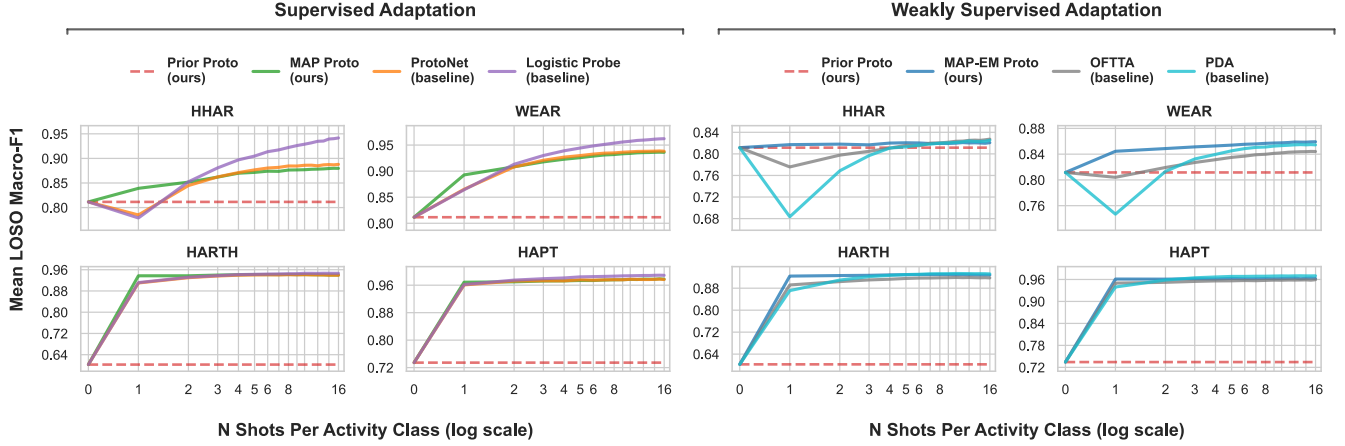


Figure 4: Mean LOSO Macro-F1 across support set sizes for supervised and weakly supervised adaptation baselines.

between embedding dimensions, they would require estimating, storing, and inverting $d \times d$ matrices, whereas our restrictions reduce adaptation to lightweight vector operations.

Table 3: Measured cost (excluding embedding extraction) in MFLOPs with $d = 32$, $T = 1$, and all classes adapted.

Dataset	K	1-shot		16-shot		TinierHAR
		MAP	MAP-EM	MAP	MAP-EM	
HHAR	6	0.002	0.008	0.014	0.104	0.586
HAPT	12	0.003	0.028	0.028	0.400	0.750
HARTH	12	0.003	0.028	0.028	0.400	0.870
WEAR	18	0.005	0.060	0.042	0.889	1.079

6 Discussion

The one-shot results demonstrate the instability of purely empirical adaptation under extremely limited calibration. Standard ProtoNets, logistic probing, OFTTA, and PDA each fall below the prior-prototype zero-shot baseline on at least one dataset, indicating sensitivity to noisy support estimates or unreliable pseudo-labels. In contrast, MAP and MAP-EM consistently improve upon the baseline using only one labeled or weakly labeled support example (3 seconds of data) per class. By combining target-user evidence with source-domain prototype priors, they reduce the risk of low-data degradation. As the support set grows to 16 shots per class, target-domain statistics become more reliable, the influence of the prior decreases, and empirical baselines approach the proposed methods. This is the intended behavior of the prior: it regularizes uncertain estimates in the ultra-low-shot regime while progressively yielding to user-specific evidence. Although logistic probing performs strongly with sufficient labeled data, its reliance on labels and gradient-based optimization limits its suitability for lightweight on-device adaptation. In contrast, our framework provides gradient-free adaptation across zero-shot, supervised, and weakly supervised calibration settings, with low computational overhead after embedding extraction. This combination of robustness, flexibility, and efficiency makes it well suited to resource-constrained wearable devices.

Several limitations remain. First, MAP-EM is currently weakly rather than fully unsupervised, since it requires knowledge of which activity classes were performed during calibration to determine which GMM components to adapt. Future work should infer the activity classes directly from the calibration data, enabling fully unsupervised adaptation. Second, although adaptation itself is source-free, repurposing a pretrained HAR classifier is not, because the prior prototypes are obtained through a one-time offline pass over the original training data. Deriving equivalent priors directly from the pretrained classifier weights would remove this dependency. Third, MAP and MAP-EM use uncertainty estimates to regularize adaptation, but they remain plug-in prototype estimators that use the ProtoNet decision rule for prediction. Future work should extend the framework to uncertainty-aware prediction through full Bayesian inference over the predictive distribution.

7 Conclusion

We presented BayaHAR, a lightweight, gradient-free framework for user adaptation in HAR on resource-constrained wearable devices. By repurposing pretrained classifiers as ProtoNets and updating their prototypes with MAP and MAP-EM, BayaHAR supports zero-shot, supervised few-shot, and weakly supervised few-shot adaptation within a single model. Across four HAR datasets, prior prototypes preserve zero-shot performance, while both methods already improve reliably from one shot (3 seconds of calibration) per activity, avoiding the early-stage degradation observed in empirical baselines. With 16 shots (48 seconds) per activity, MAP achieves macro-F1 gains of up to +33.7 pp, while MAP-EM reaches up to +32.5 pp. Since adaptation requires only closed-form prototype updates, with MAP-EM using a single EM step, it is well suited for practical on-device personalization of preexisting HAR classifiers.

Acknowledgments

This work was supported by the IPAI Foundation gGmbH (Science Residency Program) and the HammerHAI project, which received funding from the EuroHPC JU (grant No. 101234027) co-funded by the European Commission, the BMFT, and the State Ministries of Science of Baden-Württemberg, Bavaria, and Lower Saxony.

References

- [1] Rebecca Adaimi and Edison Thomaz. 2022. Lifelong Adaptive Machine Learning for Sensor-Based Human Activity Recognition Using Prototypical Networks. *Sensors* 22, 18 (Sept. 2022), 6881. doi:10.3390/s22186881
- [2] Sizhen Bian, Mengxi Liu, Lala Shakti Swarup Ray, Bo Zhou, Bin Guo, Zhiwen Yu, Thomas Plötz, Paul Lukowicz, Siyu Yuan, and Vitor Fortes Rey. 2026. Foundation Models Defining A New Era In Sensor-based Human Activity Recognition: A Survey And Outlook. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 10, 2 (2026), 1–38.
- [3] Sizhen Bian, Mengxi Liu, Vitor Fortes Rey, Daniel Geißler, and Paul Lukowicz. 2025. TinierHAR: Towards Ultra-Lightweight Deep Learning Models for Efficient Human Activity Recognition on Edge Devices. In *Proceedings of the 2025 ACM International Symposium on Wearable Computers* (New York, NY, USA, 2025-10-07) (ISWC '25). Association for Computing Machinery, 163–169. doi:10.1145/3715071.3750410
- [4] Christopher M Bishop and Nasser M Nasrabadi. 2006. *Pattern recognition and machine learning*. Vol. 4. Springer.
- [5] Marius Bock, Hilde Kuehne, Kristof Van Laerhoven, and Michael Moeller. 2024. WEAR: An Outdoor Sports Dataset for Wearable and Egocentric Activity Recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 8, 4 (Nov. 2024), 175:1–175:21. doi:10.1145/3699776
- [6] Ondrej Bohdal, Da Li, and Timothy Hospedales. 2023. Feed-Forward Source-Free Domain Adaptation via Class Prototypes. *arXiv:2307.10787 [cs.CV]* doi:10.48550/arXiv.2307.10787
- [7] Maximilian Burzer, Tobias King, Till Riedel, Michael Beigl, and Tobias Röddiger. 2025. WHAR Datasets: An Open Source Library for Wearable Human Activity Recognition. In *Companion of the 2025 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 1315–1322.
- [8] Dongzhou Cheng, Lei Zhang, Can Bu, Xing Wang, Hao Wu, and Aiguo Song. 2023. ProtoHAR: Prototype Guided Personalized Federated Learning for Human Activity Recognition. *IEEE Journal of Biomedical and Health Informatics* 27, 8 (Aug. 2023), 3900–3911. doi:10.1109/JBHI.2023.3275438
- [9] Sourish Gunesh Dhekane and Thomas Plötz. 2025. Transfer learning in sensor-based human activity recognition: A survey. *Comput. Surveys* 57, 8 (2025), 1–39.
- [10] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*. PMLR, 1126–1135.
- [11] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *Journal of machine learning research* 17, 59 (2016), 1–35.
- [12] Md Amran Hossen and Pg Emeroylariffion Abas. 2025. Machine learning for human activity recognition: State-of-the-art techniques and emerging trends. *Journal of Imaging* 11, 3 (2025), 91.
- [13] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. 2022. Lora: Low-rank adaptation of large language models. *Iclr* 1, 2 (2022), 3.
- [14] Siyu Hu, Jiqiang Liu, Chenxin Zhang, Xiaoqiang Zhu, and Lingkun Li. 2024. Proto-CSNet: A Prototype Network Model Integrating CNN and Self-Attention for Enhanced Human Activity Recognition. In *2024 20th International Conference on Mobility, Sensing and Networking (MSN)*. 48–56. doi:10.1109/MSN63567.2024.00018
- [15] Lei Jiang, Yongzhao Zhan, Zhen Jiang, and Na Tang. 2024. A Dual-Prototype Network Combining Query-Specific and Class-Specific Attentive Learning for Few-Shot Action Recognition. *Neurocomputing* 598 (Sept. 2024), 127819. doi:10.1016/j.neucom.2024.127819
- [16] Steinar Laenen and Luca Bertinetto. 2021. On episodes, prototypical networks, and few-shot learning. *Advances in Neural Information Processing Systems* 34 (2021), 24581–24592.
- [17] YiSong Li, LiWei Zou, MingXing Nie, Tao Zhu, YuanLong Wu, Yang Xiao, and KaiWen Luo. 2026. Bridging Domain and Instance Gaps: A Prototype Contrastive Framework for Robust Human Activity Recognition. *Neurocomputing* 673 (April 2026), 132868. doi:10.1016/j.neucom.2026.132868
- [18] Jian Liang, Dapeng Hu, and Jiashi Feng. 2020. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International conference on machine learning*. PMLR, 6028–6039.
- [19] Gyuyeon Lim and Myung-Kyu Yi. 2026. Prototype-Guided Physics-Modulated Perceiver for Human Activity Recognition. *IEEE Internet of Things Journal* 13, 8 (April 2026), 16117–16130. doi:10.1109/IJOT.2026.3658863
- [20] Ji Lin, Ligeng Zhu, Wei-Ming Chen, Wei-Chen Wang, and Song Han. 2023. Tiny machine learning: Progress and futures [feature]. *IEEE Circuits and Systems Magazine* 23, 3 (2023), 8–34.
- [21] Aleksey Logaciov, Kerstin Bach, Atle Kongsvald, Hilde Bremseth Bårdstu, and Paul Jarle Mork. 2021. HARTH: A Human Activity Recognition Dataset for Machine Learning. *Sensors* 21, 23 (Nov. 2021), 7853. doi:10.3390/s21237853
- [22] Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101* (2017).
- [23] Alexander Murphy, Michal Danilowski, Soumyajit Chatterjee, and Abhirup Ghosh. 2026. NEO: No-Optimization Test-Time Adaptation through Latent Re-Centering. *arXiv:2510.05635 [cs.LG]* doi:10.48550/arXiv.2510.05635
- [24] Kevin P Murphy. 2007. Conjugate Bayesian analysis of the Gaussian distribution. *def 1, 2σ2* (2007), 16.
- [25] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Minghui Tan. 2022. Efficient test-time model adaptation without forgetting. In *International conference on machine learning*. PMLR, 16888–16905.
- [26] PyTorch Team. 2026. ExecuTorch: On-Device AI Across Mobile, Embedded, and Edge for PyTorch. <https://github.com/pytorch/executorch>. GitHub repository, Accessed: 2026-05-20.
- [27] Minghui Qiu, Jun Chen, Lin Chen, Shuxin Zhong, Yandao Huang, Lu Wang, and Kaishun Wu. 2026. CODA: A Continuous Online Evolve Framework for Deploying HAR Sensing Systems. In *2026 22nd Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 19–27.
- [28] Harun Ur Rashid and Seong Ho Jeong. 2026. Federated Few-Shot Prototypical Learning for Personalized WiFi-Based Human Activity Recognition. *IEEE Access* 14 (2026), 53654–53666. doi:10.1109/ACCESS.2026.3678918
- [29] Jorge-L. Reyes-Ortiz, Luca Oneto, Albert Samà, Xavier Parra, and Davide Anguita. 2016. Transition-Aware Human Activity Recognition Using Smartphones. *Neurocomputing* 171 (Jan. 2016), 754–767. doi:10.1016/j.neucom.2015.07.085
- [30] Taoran Sheng and Manfred Huber. 2025. Reducing label dependency in human activity recognition with wearables: From supervised learning to novel weakly self-supervised approaches. *Sensors* 25, 13 (2025), 4032.
- [31] Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. *Advances in neural information processing systems* 30 (2017).
- [32] Mengyuan Song, Siwei Feng, and Tao Deng. 2026. Prototype-Guided Pseudo-Labeling for Semi-Supervised Federated Human Activity Recognition. *Pattern Recognition* 179 (Nov. 2026), 113760. doi:10.1016/j.patcog.2026.113760
- [33] Allan Stisen, Henrik Blunck, Sourav Bhattacharya, Thor Siiger Prentow, Mikkel Baun Kjærgaard, Anind Dey, Tobias Sonne, and Mads Møller Jensen. 2015. 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*. ACM, Seoul South Korea, 127–140. doi:10.1145/2809695.2809718
- [34] Baochen Sun and Kate Saenko. 2016. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*. Springer, 443–450.
- [35] TensorFlow Team. 2026. TensorFlow Lite. <https://github.com/tensorflow/tensorflow/tree/master/tensorflow/lite>. GitHub repository, Accessed: 2026-05-20.
- [36] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).
- [37] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. 2020. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726* (2020).
- [38] Shuoyuan Wang, Jindong Wang, HuaJun Xi, Bob Zhang, Lei Zhang, and Hongxin Wei. 2024. Optimization-Free Test-Time Adaptation for Cross-Person Activity Recognition. *arXiv:2310.18562 [cs.CV]* doi:10.48550/arXiv.2310.18562
- [39] Mouxiang Yang, Yunfan Li, Changqing Zhang, Peng Hu, and Xi Peng. 2024. Test-time adaptation against multi-modal reliability bias. In *The twelfth international conference on learning representations*.
- [40] Minglei Yuan, Wenhui Wang, Tao Wang, Chunhao Cai, Qian Xu, and Tong Lu. 2021. Learning class-level prototypes for few-shot learning. *arXiv preprint arXiv:2108.11072* (2021).