Human-in-the-loop Incremental Learning for Human Activity Recognition with Multimodal Wearable Sensors

HA LE, Northeastern University, USA
AKSHAT CHOUBE, Northeastern University, USA
PRANJAL KANEL, Northeastern University, USA
VARUN MISHRA, Northeastern University, USA
STEPHEN INTILLE, Northeastern University, USA

Human Activity Recognition (HAR) systems are typically trained on fixed datasets with predefined activity labels, limiting their ability to adapt to new activities and evolving user behaviors. Incremental learning offers a promising direction by enabling models to incorporate new activities with minimal data, but few-shot updates are prone to overfitting and often fail to capture co-occurring activities in real-world settings. We present a human-in-the-loop incremental learning framework for HAR that allows users to add new activities through few-shot interactions. Our approach incorporates user feedback—including co-occurrence confirmation for targeted retraining, sensor relevance, and mutual exclusivity constraints—to guide model adaptation and support multi-label recognition. We evaluate our framework on four publicly available datasets and show that it outperforms standard few-shot baselines by 5–11% in macro F1 score. Our findings highlight the value of human feedback in constraining model learning and enabling more robust adaptation. We discuss implications for designing lightweight, human-centered HAR systems. Code for the experiment is available at `/HITL-class-incremental-learning` .

ACM Reference Format:
Ha Le, Akshat Choube, Pranjal Kanel, Varun Mishra, and Stephen Intille. 2026. Human-in-the-loop Incremental Learning for Human Activity Recognition with Multimodal Wearable Sensors. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 1 (June 2026), 27 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Human Activity Recognition (HAR) aims to automatically identify daily activities from passive sensing data from wearable and mobile sensors [4, 31]. HAR is an important area in health informatics, human–computer interaction, and ubiquitous computing research, enabling applications such as long-term behavior tracking, health monitoring, and context-aware interactive systems [11, 16, 41]. Accurate and robust HAR systems can provide insights into daily routines, support personalized interventions, and enable interfaces that adapt to a user's context.

Researchers have explored machine learning and deep learning techniques for HAR. Most existing approaches follow a conventional machine learning pipeline: (1) pre-define a fixed set of activity labels, (2) collect sensor data for those activities, and (3) train a model to recognize the pre-defined activities simultaneously [18]. Researchers

Authors' Contact Information: Ha Le, le.ha1@northeastern.edu, Northeastern University, Boston, USA; Akshat Choube, choube.a@northeastern.edu, Northeastern University, Boston, USA; Pranjal Kanel, kanel.p@northeastern.edu, Northeastern University, Boston, USA; Varun Mishra, v.mishra@northeastern.edu, Northeastern University, Boston, USA; Stephen Intille, s.intille@northeastern.edu, Northeastern University, Boston, USA.

ACM 2474-9567/2026/6-ART
https://doi.org/10.1145/nnnnnnn.nnnnnnn

typically formulate this setup as a multi-class classification problem, where the model assigns each time segment to a single activity label with the highest output probability. As a result, deployed systems operate within a fixed activity corpus determined during data collection and with the assumption that activities are mutually exclusive [32].

These assumptions, however, do not hold in real-world settings. Daily routines can change as people transition through life stages (e.g., starting a new job, moving to a new home, or returning to school), and can also shift abruptly due to external factors, such as the COVID-19 pandemic [47]. These life changes often introduce new activities, remove old ones, or alter when and how frequently a person performs these activities. These circumstances may also change the context in which activities occur, e.g., working from home instead of commuting or the emergence of new hobbies, leading to shifts in behavior patterns and observed sensor signals over time. In addition, real-world activities often overlap (e.g., *"walking"* while *"talking"* or *"using phone"*), allowing multiple activities to occur simultaneously. Static HAR models trained under fixed label sets with a single output label, therefore struggle to generalize, cannot recognize new activities, and often fail to capture meaningful combinations of behaviors [74]. This limits their ability to support personalized tracking and intervention, thereby reducing their effectiveness.

Researchers have exploring the use of incremental learning and few-shot learning for HAR have explored the potential of building models that allow incorporating new activities after deployment [1, 10, 40, 79]. In these settings, users provide a small number of labeled examples and the model is retrained/fine-tuned to recognize new activities. These approaches enable more personalized and adaptable systems, but they also introduce new challenges.

With only a small set of labeled examples provided for training, models can easily overfit, especially in multimodal sensing systems with few homogeneous samples [39]. In addition, the initial activity corpus, used for pre-training, is often incomplete, leading models to learn coarse decision boundaries that group multiple similar behaviors under a single label. When new activities are introduced, these boundaries may cause confusion between existing and new classes. For example, a model trained on *"sitting," "standing," "walking,"* and *"cycling"* may associate *"sitting"* or *"standing"* with low movement; when a user performs a new activity such as *"lying down,"* the model may incorrectly classify it as *"sitting"* or *"standing."* These challenges are further exacerbated by the fact that real-world activities often co-occur (e.g., *"walking"* while *"in a conversation"*), which might cause slight alterations in the underlying signals.

To address these challenges, we frame the addition of new activities not merely as a recognition task but as an opportunity to iteratively refine and improve the underlying model. Samples from a new activity can provide positive signals for the new class and negative signals for existing classes, helping refinement of overgeneralized decision boundaries [75]. For instance, if a model initially confuses *"sitting"* with other low-movement behaviors, introducing a new activity such as *"lying down"* can provide informative negative signals that help refine the boundary of the *"sitting"* classifier. Furthermore, we can be deliberate about which classes to re-train. Although a simple solution is to retrain all classifiers on observing new labels, this is resource and time-intensive and may unnecessarily update unrelated classes, degrading their performance.

To address the aforementioned challenges, in this work, we introduce a novel incremental learning framework that leverages human feedback to guide the model update process for activity recognition. When the user adds a new activity and provides a small amount of training data in our HAR model, the framework first identifies which existing classes are "activated" on the new samples and prompt the user to confirm whether those activities co-occurred. For example, when adding a new activity such as *"typing,"* the system may detect that the *"standing"* class is also activated. It can then ask the user whether *"typing"* was performed while *"standing"*. If confirmed, these new samples are used as positive samples for the *"standing"* classifier; if rejected, they are used as negative samples. This allows the system to update only the affected classes, improving efficiency while making adaptation more interpretable and traceable. Beyond updating target classes, human feedback can also

provide lightweight domain knowledge, such as sensor relevance (which sensor is important for which activity) or activity relationships (which activities can never co-occur), to better initialize learning and reduce overfitting to noisy real-world signals.

Our framework allows users to add new activities using a few-shot data-collection process while simultaneously refining predictions for some of the existing classes. Additionally, this framework incorporates user-provided domain knowledge, such as sensor relevance and mutual exclusion constraints, to guide model updates. By updating only the relevant model parameters, our approach enables efficient, practical adaptation without requiring full retraining while effectively handling challenges with finetuning (such as small data size, co-occurrence of activities). We evaluate the proposed framework on multiple datasets and show that it improves performance over standard few-shot baselines while supporting adaptable, personalized activity recognition.

In this paper, we make the following **contributions**:

- We introduce a novel human-in-the-loop incremental learning framework for HAR that supports adding new activities with user-provided data, enables detection of co-occurring activities, and incorporates user-provided domain knowledge (e.g., co-occurrence, sensor importance, and mutual exclusivity) to guide model adaptation.
- We evaluate the framework using simulations across multiple datasets, demonstrating improved performance over standard few-shot baselines and providing empirical evidence of the benefits of enabling user-guided updates.
- We present design insights and practical considerations for building human-centered, adaptive HAR systems, highlighting the opportunities and challenges of integrating user feedback for real-world deployment.

Through extensive, simulated human-in-the-loop evaluations with four publicly available datasets, we found that our incremental learning framework outperform the few-shot baselines by 5-11% in macro F1 scores. This paper empirically demonstrates the potential opportunities and benefits of our proposed framework and outlines the guidelines for a real-world human-in-the-loop deployment.

2 Related Work

Our work is built upon prior literature in activity recognition in real-world settings, human-in-the-loop HAR systems, and incremental learning.

2.1 Human activity recognition in real-world settings

Accurate HAR models that work in real-world settings can enable continuous detection of user behaviors from wearable and mobile sensors, supporting applications in health monitoring, behavior tracking, and context-aware systems [4, 15, 19]. In real-world use, such systems can provide insights into daily routines, enable just-in-time interventions, and support personalized care [3, 28, 37].

Most HAR models, however, are trained on lab-based and pre-collected datasets, where activities are performed under controlled, clinical conditions with a fixed and limited label set [13, 54, 56]. These settings do not reflect the variability of real-world environments, including differences in user behavior, sensor placement, and context [1, 52, 64]. As a result, deployed models are largely static: they assume a fixed set of activities and do not adapt as behaviors change over time.

This mismatch between controlled training data and real-world usage leads to several challenges. First, models degrade in performance when encountering distribution shifts in routines, environments, or sensing conditions [74]. Second, many activities that are meaningful for a specific individual may not be included in the predefined label set, creating blind spots in the model's predictions. Third, even for known activities, real-world behavior is more diverse than what is captured in lab settings [34, 36]. While lab data typically reflect clean, isolated executions of activities, real-world behaviors vary across individuals and contexts, often consisting of

multiple sub-activities or different ways of performing the same task [32]. Furthermore, activities frequently co-occur (e.g., walking while using a phone or talking), and these combinations can alter the underlying sensor signals. Because most datasets treat activities as mutually exclusive and do not capture such variability or interactions, models trained on them often struggle to generalize, leading to misclassifications or failure to recognize meaningful distinctions in real-world usage.

2.2 The need for human-in-the-loop, feedback-driven HAR systems

The challenges described above highlight the need for human-in-the-loop, feedback-driven HAR systems that can adapt to individuals and evolving real-world conditions. Prior researchers have extensively explored collecting self-reported behavioral and activity labels in daily life, primarily focusing on efficient ways to gather annotations from participants [20, 25]. These approaches generally fall into two categories: retrospective recall and in-situ reporting. Retrospective recall asks participants to reconstruct their activities after the fact, often supported by contextual cues such as sensor data [35, 42, 65]. While this can produce detailed annotations, it imposes substantial cognitive burden and requires significant time and effort [35].

In contrast, in-situ methods collect labels when the activity is happening. A widely used approach is ecological momentary assessment (EMA), in which participants are prompted throughout the day to report their current activity or context [60]. EMA reduces recall bias and distributes effort over time, but frequent interruptions can be burdensome and may lead to missing data [46, 49]. More recently, micro-EMA (μEMA) reduces interaction burden by using only brief, low-effort prompts that can be sustained at higher frequency [23, 51]. The original smartwatch-based design of μEMA, however, relies on multiple-choice questions, limiting the ability to capture activities beyond a predefined label set [50]. Recent open-ended, speech-based, and multimodal μEMA approaches, enabling the collection of a broader and more personalized set of activity and contextual labels [27, 34, 36].

Beyond label collection, human-in-the-loop interactions can provide richer feedback to guide model learning. Prior efforts have largely focused on querying users about uncertain or unknown activities during inference [14, 30, 61], with less attention to incorporating participants' domain knowledge into training [3, 37, 67]. This is especially important in real-world deployments, where models must adapt from limited data. In such settings, few-shot learning is prone to overfitting, because models may rely on homogeneous patterns rather than the true underlying signal [39]. User feedback can mitigate this issue by providing constraints or context about what defines an activity, guiding the model toward the correct signal [5, 26, 71]. By narrowing the hypothesis space and reinforcing meaningful patterns, such feedback might improve generalization and enable faster, more reliable adaptation without requiring extensive labeled data.

2.3 Incremental learning for HAR

Incremental learning refers to updating a model as new data, users, or activity classes become available, rather than retraining from scratch [72, 81]. This is particularly relevant to HAR, where activity vocabularies evolve over time, user behaviors vary, and collecting large labeled datasets for each update is impractical. As a result, incremental learning has been explored to support adaptation while reducing annotation and retraining cost.

Researchers have proposed a range of approaches for incremental HAR. Early methods detect and incorporate unseen activities by updating model structures over time [48]. Subsequent work has focused on mitigating catastrophic forgetting and supporting multimodal inputs, for example through regularization-based methods and representation alignment across sensors [79]. Other approaches emphasize personalization, enabling continuous adaptation to new users without requiring labeled target data [43], or leverage semantic representations and active learning to recognize novel activities with minimal user input [10]. More recent work explores continual learning frameworks based on prototype learning and replay mechanisms for streaming data, as well as architectures that expand to accommodate heterogeneous sensing modalities [1, 40].

Researchers have focused on techniques to reduce catastrophic forgetting [57, 59], supportpersonalization [43], or recognize new class under single-label settings. Comparatively little research has been conducted on incremental learning for co-occurring, multi-label activities, where multiple behaviors may occur simultaneously and updates must account for overlaps between existing and newly added classes. Moreover, with the partial exception of approaches that query users under uncertainty, researchers have not explicitly studied how human-in-the-loop interaction can guide model updates by incorporating user knowledge. In this work, we introduce an incremental HAR framework with human-in-the-loop components to support expanding activity corpus, handle multilabel recognition, and minimize retraining at each update step.

3 Human-in-the-loop, incremental learning framework

We present the rationale and details for our framework.

3.1 Problem statement

We consider the problem of incrementally expanding a wearable-based human activity recognition system in deployment. Given a system pre-trained on a small set of seed activities $\mathcal{A}_0 = \{a_1, \ldots, a_S\}$, our aim is to extend it to recognize a growing set of activities $\mathcal{A}_t = \mathcal{A}_0 \cup \{a_{S+1}, \ldots, a_{S+t}\}$ as new activities are introduced one at a time. At each step t, a small amount of labeled data $\mathcal{D}_t$ for the new activity is provided by the user.

The system must satisfy two requirements. First, it must correctly recognize the new activity after each addition. Second, it must support multi-label prediction: in practice, activities frequently co-occur (e.g., *"opening a fridge"* while *"standing"*), and the system must recognize such combinations without requiring users to explicitly annotate every possible activity pairing.

This second requirement is particularly challenging because the number of possible co-occurring combinations grows quadratically with the size of the activity corpus, making exhaustive annotation infeasible. We instead exploit the co-occurrence structure that naturally emerges when new activities are added — using it to update predictions of existing classes without additional annotation effort and to enforce mutual-exclusion constraints that suppress physically impossible combinations at inference time.

3.2 Supporting User Feedback via Microinteractions

A core design principle of our framework is that the annotation burden on the user should be minimal with a short interaction. Instead of requiring users to label raw sensor windows or define training sets, we collect feedback using a small number of lightweight microinteractions that occur when each new activity is added. These interactions are designed to be completable in under a minute and require minimal machine learning knowledge [2, 23].

We propose three types of user feedback, each targeting a different component of the system. The interaction start with participants indicating they want to add a new activity a, and have provided a small amount of samples (e.g., 30-60 seconds of performing the activity) for training.

Co-occurrence confirmation. When a new activity is added, the system automatically runs its predictions on the new activity's data and presents the user with a ranked list of existing activities that appear to co-occur with it. For each candidate pair, the user confirms or denies the co-occurrence with a single binary response (e.g. *"Does opening the fridge happen while standing?"*). Confirmed co-occurrences trigger targeted retraining of the affected existing heads; denied ones are used as negative evidence. This interaction replaces requiring the user to collect and annotate combined activity data for every possible pair.

Sensor relevance specification. When adding a new activity, the user optionally rates the relevance of each sensor to that activity on a three-point scale: low importance, unsure, or high importance. For example, an

motion sensor stuck on the fridge door would be rated high importance for *"open fridge"* but low importance for *"walking"*. All sensors are defaulted to unsure at initialization. This interaction takes the form of a simple checklist and requires only basic familiarity with the sensor placement.

Mutual exclusion marking. The user identifies a small set of activities that cannot physically co-occur with the new activity — for example, *"walking"* and *"sitting"* are mutually exclusive. These constraints are added to a global registry and enforced at inference time to suppress physically impossible simultaneous predictions. In practice, users can identify three to five such pairs per activity in a few seconds by scanning a list of previously learned activities.

Together, these three microinteractions give the system structured prior knowledge that would be expensive to learn from data alone, while keeping the total user effort per new activity to a minimum. Figure 1 illustrates examples of each interaction type.

3.3 Framework Formulation

Our framework adopts a multi-head architecture in which a shared encoder produces a fixed embedding, and each activity $a \in \mathcal{A}_{S+t}$ is modeled by an independent binary classifier f_a. This design enables multi-label prediction, allowing multiple activities to be recognized simultaneously without needing to enumerate all possible combinations during training. It also supports targeted updates: when a new activity is introduced, only the heads whose behavior is affected by the new data needs to be updated, rather than retraining the entire model. In our framework, we did not update the encoder/representation, and only update/retrain the binary classification heads. This approach ensure that updating one activity classification head does not cause catastrophic forgetting in other activities'.

Our incremental learning framework consists of two parts: part 1 is co-occurrence detection and incremental updates to old classifiers, and part 2 is the addition of a new binary classifier based on user-specified activities.

3.3.1 Co-occurrence Detection and Incremental Updates of Existing Heads. When user introduces a new activity a_{S+t} with a small amount of few-shot samples, we first assess how existing heads respond by running inference on the new activity's samples $\mathcal{D}_t$. For each trained head h_a, we compute its mean predicted likelihood over these samples:

$$\bar{p}_a = \frac{1}{|\mathcal{D}_t|} \sum_{x \in \mathcal{D}_t} \sigma(f_a(x))$$

A head fires if $\bar{p}_a \geq 0.6 \cdot \tau_a$, where τ_a is its calibrated threshold. The factor 0.6 accounts for partial co-occurrence — if two activities co-occur in only 40% of windows, the mean probability will be lower than the full threshold even if the head is working correctly. The fired heads are compared against the ground-truth (collected from users' confirmation or rejection), yielding confirmed co-occurrences (true positives), missed co-occurrences (false negatives), and spurious firings (false positives).

For missed co-occurrences, we distinguish two cases. If the head already correctly classifies the co-occurring windows in isolation but fails to fire globally (i.e., the likelihood predicted is the highest among all classes, but not high enough to cross the threshold), we apply a small downward adjustment to its decision threshold:

$$\tau_a \leftarrow \text{clip}\left(\bar{p}_a \cdot \eta,\ \tau_{\min},\ \tau_a\right)$$

where η is a nudge factor (how aggressive we want to adjust the threshold) and $\tau_{\min}$ prevents thresholds from collapsing (empirically we set $\eta = 0.9$ and $\tau_{\min} = 0.2$). If the head genuinely misclassified the co-occurring windows, we retrain it using those windows as additional positives. For spurious firings, we retrain the head

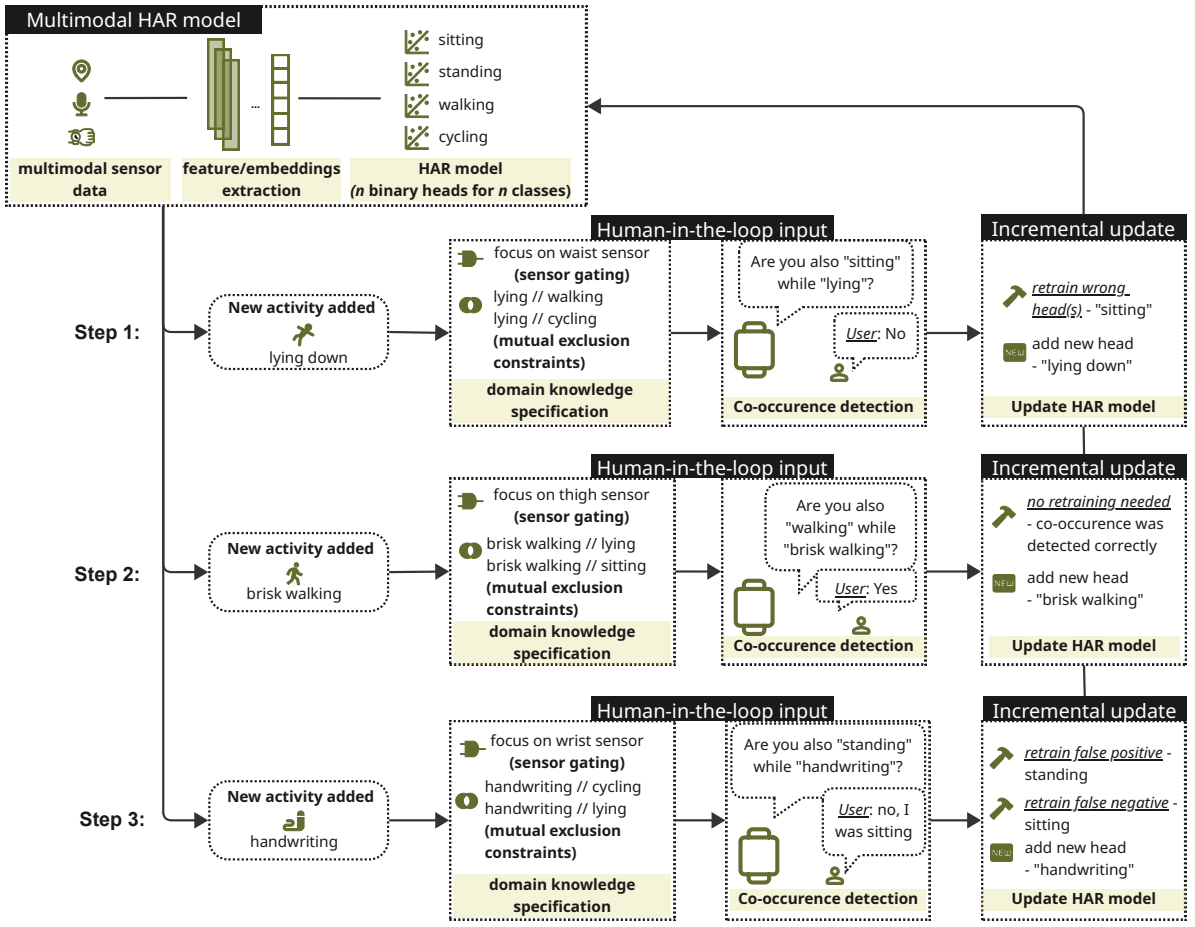


Fig. 1. Sequence of human interactions in our proposed incremental learning framework. First, the user indicates the activity they want to add and provides few-shot samples. They also provide domain knowledge about the new activity (sensor importance and mutual exclusivity with existing activities). The system then runs the new samples through the existing classifiers and asks the user to confirm or reject co-occurrence. User responses are used as training signals for the existing classifiers, while the domain knowledge is incorporated into the training of the new classifier.

using non-co-occurring windows of the new activity as additional negatives. In all retraining cases, previously seen positive examples are replayed alongside the new data to prevent forgetting.

3.3.2 Adding new classifier heads with user specifications. A new binary head is trained for a_{S+t} using its labeled training data, with two forms of user-specified domain knowledge incorporated into the head architecture.

- Sensor gatings: These are applied to the encoder embeddings before classification, focusing the head on discriminative channels and suppressing irrelevant sensor noise.

Dataset	Subjects	Multi-sensor	Free-living	Number of activities
Opportunity [13]	4	5 body motion + 12 object sensors	×	4 locomotion labels + 11 fine-grained gestures
PAAWS Lab [52]	20	5 body motion sensors	×	41 activities
PAAWS FL [52]	20	8 body motion sensors	✓	55 activities
WEAR [6]	23	4 body motion sensors	×	19 activities

Table 1. Datasets used for testing.

- Mutual exclusivity (ME) constraints: The user identifies a set $\mathcal{M}(a_{S+t})$ of activities that cannot co-occur with the new activity. These pairs are added to the global mutual exclusion registry $\mathcal{M}$, which is applied as a post-hoc suppression step during inference.

4 Experiment protocol

In this section, we describe the evaluation conditions for our proposed framework: the datasets, the baseline and ablation conditions, and how we simulate human-in-the-loop interactions. Since our framework is designed to support personalized modeling and definition of activities, we conduct within-subject, few-shot learning evaluations. For each participant in these dataset, we held out a portion of the dataset as train/val set (80/20 split), and the rest of the data is held out as test set. We included the details about train and test set for each dataset in the subsection below.

To demonstrate the utility of our framework in a multi-sensor and multi-label setting, we simulate evaluation scenarios on four different datasets.

4.1 Datasets

We selected datasets with the following properties: 1) contains multiple sensors, and 2) has a large set of labels (especially with multiple labeling schemes or a multilabel annotation scheme).

4.1.1 Opportunity Dataset [13]. This is a publicly available HAR dataset with five body motion sensors and 12 object motion sensors recorded while four subjects executed daily living activities. Each participant performed six different runs, including five activity of daily living (ADL) runs and a drill run. The drill run was designed such that participants performed the activities in a more constrained scenario, while the ADL runs consisted of temporally unfolding situations. The dataset contains labels for high-level behaviors, locomotion, and gestures. In this experiment, we used the locomotion (4) and gesture labels (11) for our simulation. For each participant in the dataset, we used the drill run as the train/val set and the five ADL runs as the test set.

4.1.2 Physical Activity Assessment Using Wearable Sensors (PAAWS) Dataset - Lab [52]. The PAAWS Lab dataset contains recordings of 41 physical activities collected using wearable accelerometer sensors. Data were recorded using ActiGraph GT9X Link sensors at a sampling rate of 80 Hz with a measurement range of ±8g. During data collection, participants wore 20 accelerometer sensors placed on both the left and right sides of the body. The sensors were positioned as follows: four sensors around each ankle (lateral, anterior, posterior, and medial positions), one sensor in the middle of each thigh, three sensors on each side of the waist (anterior axillary line, midaxillary line, and posterior axillary line), and two sensors on each wrist (one on the top of the wrist, similar to where a watch is commonly worn, and one on the underside of the wrist). For our experiments, we used eight out of 20 sensors to simulate a simpler configuration, and data from 20 participants[1] in this dataset. Specifically, we used accelerometer data from the anterior ankle, thigh, midaxillary waist, and top wrist sensors on both sides

[1]For our experiments, we use data from participants with IDs: `DS_10–DS_29`.

of the body. For each activity with longer than 3 min of readings, we used the first minute as the train/val set, and the rest as the test set. For activities with less than 3 min of readings, we used a third of the data as the train/val set, and the rest as held out the test set.

4.1.3 Physical Activity Assessment Using Wearable Sensors (PAAWS) Dataset - Free-living [52]. The PAAWS FL dataset is a seven-day, unsupervised free-living dataset contains data from 20 participants[2] with the recordings of five ActiGraph GT9X Link sensors placed on the tops of the left and right wrists (like a watch), on the right waist, in the middle of the right thigh, and the right ankle. It contains 55 unique activity labels, however, different participants in the dataset had different set of activities, ranging from 15 to 26 activities. For each activity with longer than 15 min readings, we used the first five minutes as train/val set, and the rest as test set. For activities with less than 15 min of human-annotated annotation from front-facing camera, we used a third of the data as the train/val set, and the rest as the held out test set.

4.1.4 WEAR Dataset [6]. WEAR is an outdoor sports dataset for both vision- and motion-based human activity recognition (HAR). Data from 23 participants performing a total of 18 different workout activities were collected with synchronized accelerometer and camera (egocentric video) data recorded at 11 different outside locations. Accelerometer data was collected at 50 Hz with a sensitivity of ±8g using four open-source Bangle.js smartwatches running a custom, open-source firmware. For our experiments, we used the triaxial acceleration data collected from the left/right wrists and the left/right ankles. Each participant performed every activity for at least 90 seconds. We used 30 seconds of data for each activity for training/validation, and the rest as held out test sets.

4.2 Baselines and Ablations

We compare the proposed framework against two baselines that bound performance under different levels of supervision and adaptation.

(1) Online few-shot baseline (lower bound). This setting mirrors the incremental setup but removes all human-in-the-loop components. We used the same model architecture as Full HITL, but without sensor gating (sensor weights are learned during training) and without post-inference mutual exclusivity (ME) constraints. When a new activity was introduced, only a new classifier head was trained using new samples; previously learned classifiers were left unchanged and got no updates.

(2) Full retraining baseline (upper bound). This setting used the same base architecture as Full HITL, but without sensor gating or ME constraints (similar to the lower bound baseline). Each time a new activity is added, its samples were incorporated into a full retraining procedure: all existing classifiers were updated, with the new activity samples treated as negatives for previously learned classes.

Together, these baselines provide lower and upper bounds on performance: the online few-shot baseline reflects a minimal adaptation strategy without user guidance, while the offline supervised baseline reflects a more data-intensive approach with full retraining.

We conducted ablation experiments to understand the impact of each individual component of our framework on the overall performance. These conditions use the same overall pipeline but remove a key component of the proposed framework. Specifically, it excludes (1) targeted retraining of existing heads, (2) ME constraints, and (3) sensor gating.

4.3 Simulation of human-in-the-loop interactions

We initialize the system with the same set of seed activities for every participant in each dataset (Table 2), and introduce the remaining activities incrementally. For the structured, lab-based datasets (PAAWS Lab, Opportunity,

[2]For our experiments, we use data from participants with the following IDs: `DS_10`, `DS_138`, `DS_139`, `DS_140`, `DS_235`, `DS_239`, `DS_240`, `DS_246`, `DS_249` `DS_36`, `DS_37`, `DS_38`, `DS_39`, `DS_42`, `DS_44`, `DS_48`, `DS_51`, `DS_58`, `DS_59`, and `DS_87`.

Dataset	Seed Activities	Added Activities	Order of adding
Opportunity	lie, sit, stand, walk	open dishwasher, close dishwasher, open fridge, close fridge, toggle switch, close drawer, open drawer, open door, close door, clean table, drink from cup	random
WEAR	jogging, lunges, sit-ups	stretching triceps, stretching hamstrings, stretching shoulders, stretching lunging, lunges complex, push-up complex, bench dips, stretching lumbar rotation, burpees, push-ups, jogging skipping, jogging butt-kicks, jogging sidesteps, sit-ups complex, jogging rotating arms	random
PAAWS (Lab)	cycling, sitting still, standing still, walking	conversation, loading shelf, playing frisbee, vacuuming, washing dishes, sitting with movement, organizing shelf cabinet, reclining, web browsing, unloading shelf, folding clothes, chopping food, writing, treadmill (3 mph), treadmill (3 mph free walk), standing with movement, machine chest press, ab crunches, lying on stomach, arm curls, treadmill (3 mph with briefcase), treadmill (3 mph using phone), typing, walk downstairs, push-ups, walk upstairs, treadmill (3 mph drinking), lying on back, lying on left side, treadmill (2 mph), lying on right side, stationary bike, treadmill (4 mph), treadmill (5.5 mph)	random
PAAWS (FL)	cycling, sitting still, standing still, walking	applying makeup, blow-drying hair, brushing teeth, stationary bike, resistance training (free-weight), resistance training, dry mopping, dusting, flossing teeth, folding clothes, ironing, kneeling still, kneeling with movement, lying still, lying with movement, organizing shelf, playing frisbee, putting clothes away, sitting with movement, sweeping, vacuuming, walking downstairs, walking fast, walking slow, walking treadmill, walking upstairs, washing face, washing hands, watering plants	in dataset order (per participant)

Table 2. Summary of seed and incrementally added activities across datasets.

and WEAR), where activities are performed under researcher instructions and do not follow a natural temporal order, we randomize the sequence in which activities are introduced for each participant. In contrast, for the PAAWS FL dataset, which consists of free-living, naturally occurring behaviors, we preserve the original order of activity appearance in the annotations to reflect their real-world temporal progression.

Sensor relevance scores and mutual exclusion constraints are defined once per dataset by a single researcher. For each activity, the researcher assigns a tag of "high importance", "low importance" and "unsure" to each sensor channel indicating its expected relevance. For example, a fridge accelerometer was tagged as "high importance" for *"open fridge"* but "low importance" for *"walking"*. The researcher also identifies a set of activities (from three to five activities) that cannot physically co-occur with each activity, such as *"walking"* and *"sitting"*, which are used to suppress conflicting predictions at inference time. We include details of the simulation in the supplemental materials.

4.4 Evaluation Metrics

We report macro and weighted F1 metrics at every step (a step is when a new activity is added to the model), as well as the final metrics after all activities have been added. We report both metrics because they capture

complementary aspects of performance in our incremental setting. Because activity classes are inherently imbalanced — locomotion classes such as *"walking"* and *"sitting"* account for substantially more windows than short-duration gestures such as *"toggle switch"* or *"drink from cup"* — weighted F1 reflects overall system performance weighted by real-world activity frequency. Macro F1 treats all activities equally regardless of frequency, making it sensitive to whether rare gesture classes are being recognized correctly. Together, the two metrics allow us to assess both the practical utility of the system and its ability to recognize infrequent activities that are the primary target of the incremental learning process.

We report F1 separately for three subsets of activities at each step: *seed F1* measures performance on the four initial locomotion activities, tracking whether the human-in-the-loop (HITL) process degrades previously learned knowledge; *new activity F1* measured performance on activities added through the incremental loop, reflecting how well the system learns each new class; and *all F1* aggregated performance across the complete activity set at that step (seed and added activities). We also report the total number of retraining events triggered across the incremental process, where a retraining event refers to an update to an existing head's weights in response to a co-occurrence detection — as opposed to threshold adjustments or training a new head from scratch.

4.5 Framework Implementation

We implemented our incremental learning framework consistently across all four datasets, using accelerometer data resampled to 20Hz and segmented with a 5-second sliding window and one-second stride. All models were implemented in PyTorch and trained on a local NVIDIA RTX 5090 GPU. Code for all experiments will be made available at `/HITL-class-incremental-learning` upon publication.

The design choices described below (e.g., encoder, window size, ME and sensor gating mechanism) represent one concrete implementation of the framework used for evaluation. They are meant to illustrate how the system can be implemented in practice, rather than to exhaustively explore or optimize all possible configurations. The framework can be implemented with alternative encoders (e.g., masked autoencoder [17], STMAE [45]), representations, binary classifier (e.g., random forest, prototype-based [1]), experience replay [58], or more sophisticated sensor gating or ME constraints methods.

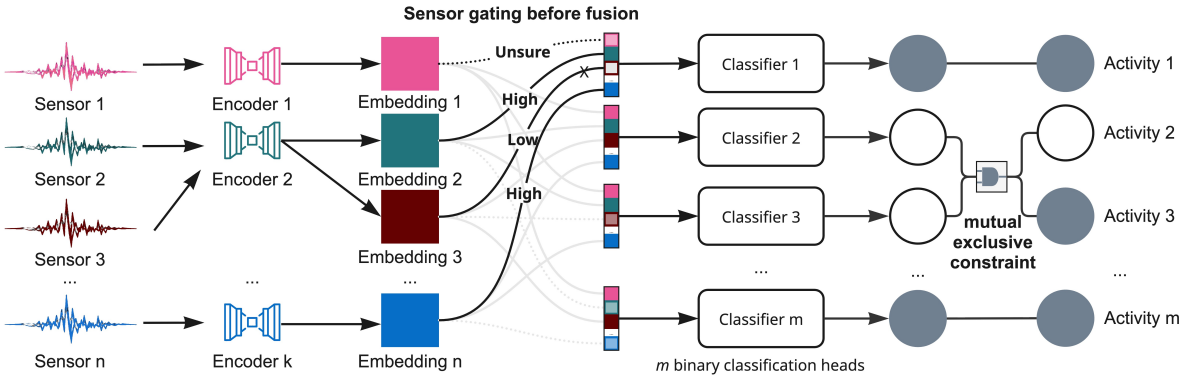
4.5.1 Pre-trained Encoder. We pre-trained a SimCLR encoder on approximately 800 hours of accelerometer data from 20 participants in the PAAWS Lab dataset[3]. These are completely separate from the 20 participants we include in the incremental learning experiments. The encoder follows the TPN architecture of Saeed et al. [55] — three stacked 1D convolutional layers (32, 64, and 96 filters with kernel sizes 24, 16, and 8 respectively), each followed by 10% dropout, with a global max pooling layer producing a 96-dimensional embedding per window. A three-layer projection head is attached during contrastive training and discarded afterwards. Training uses the NT-Xent loss [8] with four augmentations: time warp, rotation, Gaussian noise, and amplitude scaling.

4.5.2 Binary Classifier Heads. Each activity a is represented by a dedicated binary classification head f_a operating on frozen encoder embeddings. Let L denote the number of sensors and D the embedding dimension.

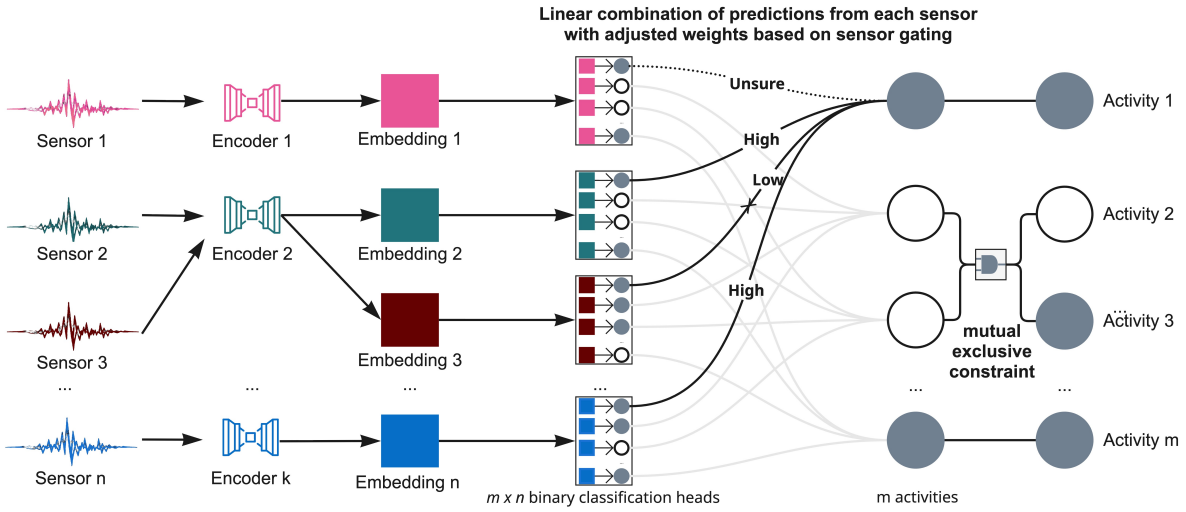
Early fusion (see Figure 2a). We extract an embedding from each sensor independently using a shared encoder, resulting in an embedding matrix $Z \in \mathbb{R}^{n \times D}$, where n is the number of sensors and D is the embedding dimension. To incorporate user-provided sensor relevance, we apply a per-sensor gate $g \in \{0, 0.5, 1.0\}^n$, in which 0 = "low importance", 0.5 = "unsure" and 1 = "high importance", which scales each sensor's embedding:

$$\tilde{Z} = Z \odot g$$

[3]For pre-training the encoder, we use accelerometer data from participants with IDs `DS_30`–`DS_39` in the PAAWS Lab dataset on the same eight sensors listed in Section 4.1



(a) Early fusion.



(b) Late fusion.

Fig. 2. Architecture of our framework implementation for early (a) and late (b) fusion. Each sensor modality (e.g., IMU, audio, PPG, camera) is processed by a corresponding encoder to extract embeddings. Sensor streams with similar characteristics (e.g., accelerometers from different devices) can share the same encoder. In our experiments, all inputs are accelerometer signals collected from different body locations or objects; thus, we use a single SimCLR encoder pretrained on multi-limb ActiGraph data.

The gated embeddings are then flattened into a single vector and passed through a two-layer MLP to produce a binary prediction:

$$\hat{y} = f_{\text{a}}(\tilde{Z}_{\text{flat}})$$

where $\tilde{Z}_{\text{flat}} \in \mathbb{R}^{n \times D}$ and f_{a} consists of a linear layer with 256 units, followed by ReLU activation and 0.3 dropout.

Late fusion (see Figure 2b). In contrast, late fusion processes each sensor independently. For each sensor $l \in \{1, \ldots, n\}$, we apply a separate binary classifier to its embedding $Z_l \in \mathbb{R}^D$, scaled by the corresponding gate value g_l. The resulting per-sensor predictions are then combined using a normalized weighted sum:

$$\hat{y}_l = f_a^{(l)}(Z_l \cdot g_l) \qquad \hat{y} = \frac{\sum_{l=1}^{n} w_l \cdot \hat{y}_l}{\sum_{l=1}^{n} w_l}$$

Training. All classifier heads are trained with focal loss ($\gamma = 2$) to handle class imbalance, with positive class weight capped at 10. We use Adam [29] with learning rate 10^{-3} for new heads and 10^{-4} for retraining, with early stopping on validation AUC (patience 10). Decision thresholds are calibrated per head on the validation set by maximizing F1 on the precision-recall curve, constrained to $[0.2, 0.8]$ to avoid collapse.

At inference time, ME constraints are enforced as a post-hoc suppression step. Let $\mathcal{M} = \{(a, b)\}$ be the set of mutually exclusive activity pairs. For each firing pair $(a, b) \in \mathcal{M}$, we suppress the lower-confidence prediction:

$$\hat{y}_a = \begin{cases} 0 & \text{if } \exists\, b : (a, b) \in \mathcal{M},\ p_b > p_a \\ \mathbb{1}[p_a \geq \tau_a] & \text{otherwise} \end{cases} \tag{1}$$

where $p_a = \sigma(l_a)$ is the predicted probability and τ_a is the calibrated threshold for activity a.

5 Evaluation results

In this section, we present the results from our experiments. Our findings include a direct comparison with baseline, analysis of the class boundaries, and an analysis of the impact of ME, sensor gating, and seed activity choice.

5.1 Comparisons with baselines

Our framework outperforms the few-shot baseline in all datasets in both macro and weighted F1 scores, with significantly fewer retraining episodes compared to the full retraining condition. Table 3 provides detailed performance breakdowns for seed activities, newly added activities, and all activities combined.

For seed activities (see Table 3a), Full HITL consistently preserves or improves baseline performance, with the most pronounced macro F1 improvements observed on WEAR (early fusion: ▲ + .69; late fusion: ▲ + .49), PAAWS FL (early: ▲ + .20; late: ▲ + .16), and PAAWS Lab (early: ▲ + .16; late: ▲ + .11). These gains are driven almost entirely by the targeted retraining mechanism, as the *w/o retrain* condition is close to the baseline numbers across all datasets.

For newly added activities (Table 3b), improvements in macro F1 over the baseline are more modest. Full HITL yields gains on Opportunity (▲ + .04) and PAAWS Lab (▲ + .02), while performance remains largely stable on PAAWS FL (▲ + .01) and WEAR. These trends are influenced by the choice of seed activities, as well as the characteristics of the activities and sensor configurations, which we examine in more detail in Section 5.5.

For all activities performance (see Table 3c), Full HITL yields consistent improvements over the baseline on every dataset, with improvements in macro F1 scores ranging from ▲ + .03 on PAAWS Lab and ▲ + .12 on WEAR. The average number of retraining episodes per run is small relative to the total number of activities added (see Table 3d) — with the exception of 46.2 ± 12.3 retrain episodes on WEAR dataset under late fusion (still small compared to full retrain with 165 episodes) — indicating that the co-occurrence detection mechanism selectively triggers retraining only when necessary. Looking at aggregated results, sensor gating and mutual exclusion contribute marginal and inconsistent effects across datasets. We take a closer look at the impact of sensor gating and ME constraints on individual activity in Section 5.3 and 5.4.

Table 3. Performance comparison across datasets with ablation analysis (mean ± SD). We **highlight** conditions where our full framework outperformed the few-shot baseline.

Fusion	Condition	PAAWS (Lab)		PAAWS (FL)		Opportunity		WEAR	
		M F1	W F1	M F1	W F1	M F1	W F1	M F1	W F1
(a) Seed activities performance									
	Baseline	.61±.10	.72±.05	.32±.10	.58±.20	.65±.03	.69±.04	.19±.07	.19±.07
Early	Full HITL	**.77±.07**	**.82±.04**	**.52±.12**	**.74±.12**	**.67±.04**	**.70±.04**	**.88±.08**	**.88±.08**
	w/o ME	.77±.07	.82±.03	.52±.12	.73±.12	.67±.04	.70±.04	.89±.08	.88±.09
	w/o retrain	.61±.10	.72±.05	.32±.10	.58±.20	.65±.03	.69±.04	.19±.07	.19±.07
	w/o sensor gating	.78±.06	.82±.03	.51±.11	.73±.11	.67±.04	.70±.04	.89±.07	.89±.07
	Full retrain	.78±.03	.85±.03	.54±.06	.79±.08	.65±.07	0.68±0.02	.92±.20	.92±.19
	Baseline	.56±.10	.68±.08	.33±.10	.60±.21	.46±.06	.57±.06	.40±.08	.40±.09
Late	Full HITL	**.67±.09**	**.77±.08**	**.49±.10**	**.73±.10**	.44±.06	.56±.06	**.89±.07**	**.89±.08**
	w/o ME	.67±.09	.76±.08	.48±.09	.73±.10	.44±.06	.56±.06	.89±.08	.89±.08
	w/o retrain	.56±.10	.68±.08	.33±.10	.60±.21	.46±.06	.57±.06	.40±.08	.40±.09
	w/o sensor gating	.67±.09	.77±.08	.48±.09	.73±.10	.45±.05	.56±.05	.90±.07	.89±.08
	Full retrain	.87±.05	.88±.06	.52±.05	.75±.06	.42±.01	.56±.07	.89±.20	.89±.19
(b) New activities performance									
	Baseline	.74±.05	.72±.04	.44±.06	.77±.08	.47±.04	.51±.05	.81±.05	.81±.05
Early	Full HITL	**.76±.03**	**.76±.03**	**.45±.08**	**.77±.08**	**.51±.07**	**.54±.05**	.81±.06	.81±.06
	w/o ME	.76±.03	.76±.03	.44±.06	.77±.08	.52±.05	.55±.05	.81±.05	.81±.05
	w/o retrain	.74±.04	.73±.04	.44±.07	.77±.08	.48±.02	.51±.04	.79±.06	.79±.06
	w/o sensor gating	.75±.05	.75±.03	.45±.06	.77±.08	.51±.03	.54±.04	.81±.06	.81±.06
	Full retrain	.78±.02	.81±.03	.48±.06	.82±.08	.45±.05	.50±.06	.90±.19	.90±.19
	Baseline	.74±.04	.72±.04	.46±.07	.77±.08	.45±.03	.48±.03	.78±.07	.78±.07
Late	Full HITL	**.76±.04**	**.75±.04**	.44±.07	**.77±.08**	**.50±.03**	**.53±.04**	.77±.07	.77±.08
	w/o ME	.76±.04	.75±.04	.45±.06	.77±.09	.53±.04	.56±.04	.76±.08	.75±.08
	w/o retrain	.74±.04	.73±.04	.45±.07	.77±.08	.47±.03	.50±.05	.74±.06	.74±.07
	w/o sensor gating	.76±.04	.74±.04	.45±.07	.77±.08	.53±.03	.56±.05	.77±.07	.77±.08
	Full retrain	.85±.12	.87±.12	.48±.06	.82±.08	.17±.02	.20±.06	.89±.19	.89±.19
(c) All activities performance									
	Baseline	.73±.05	.72±.04	.42±.05	.71±.09	.52±.03	.64±.03	.70±.05	.70±.05
Early	Full HITL	**.76±.03**	**.78±.02**	**.46±.07**	**.77±.07**	**.55±.05**	**.65±.04**	**.82±.05**	**.82±.06**
	w/o ME	.76±.03	.79±.02	.46±.06	.77±.07	.56±.04	.65±.03	.82±.05	.82±.05
	w/o retrain	.73±.04	.73±.03	.42±.06	.71±.09	.52±.02	.64±.03	.69±.05	.68±.05
	w/o sensor gating	.75±.04	.78±.02	.46±.06	.76±.07	.56±.03	.65±.04	.82±.06	.82±.06
	Full retrain	.78±.04	.81±0.3	.50±.09	.80±.08	0.51±.04	.63±.02	.90±.19	.90±.19
	Baseline	.72±.04	.71±.04	.44±.05	.72±.09	.45±.02	.54±.04	.72±.06	.72±.07
Late									

(continued on next page)

(continued)

Fusion	Condition	PAAWS (Lab)		PAAWS (FL)		Opportunity		WEAR	
		M F1	W F1	M F1	W F1	M F1	W F1	M F1	W F1
	Full HITL	**.75±.04**	**.76±.04**	**.45±.06**	**.77±.07**	**.49±.04**	**.55±.04**	**.79±.07**	**.79±.07**
	w/o ME	.75±.04	.76±.04	.45±.06	.76±.08	.51±.04	.56±.04	.78±.07	.78±.07
	w/o retrain	.72±.04	.71±.04	.43±.05	.73±.09	.47±.03	.55±.05	.68±.06	.68±.06
	w/o sensor gating	.75±.04	.75±.04	.45±.06	.76±.07	.51±.03	.56±.04	.79±.06	.79±.07
	Full retrain	.86±.04	.87±.12	.49±.09	.79±.08	.24±.02	.45±.04	.89±.20	.91±.19
(d) Average retrain episodes (number of heads getting retrained during the incremental learning process)									
	Baseline	–		–		–		–	
	Full HITL	**21.2±5.3**		**10.9±3.8**		**7.8±3.3**		**12.8±2.5**	
Early	w/o ME	20.6±5.4		11.1±3.8		6.5±2.5		13.5±3.7	
	w/o retrain	–		–		–		–	
	w/o sensor gating	21.1±5.0		10.9±3.8		9.2±2.2		12.0±2.3	
	Full retrain	851.0±0.0		250.1±78.2		110.0±0.0		165.0±0.0	
	Baseline	–		–		–		–	
	Full HITL	**22.1±4.8**		**12.1±4.7**		**17.2±2.5**		**46.2±12.3**	
Late	w/o ME	21.4±4.3		11.7±4.5		18.0±2.8		47.4±10.3	
	w/o retrain	–		–		–		–	
	w/o sensor gating	21.7±4.8		11.8±4.7		18.8±0.5		39.3±12.0	
	Full retrain	851.0±0.0		250.1±78.2		110.0±0.0		165.0±0.0	

We present the step-wise progression of macro F1 scores in Figure 3. Across all four datasets, the Full HITL framework consistently outperforms the few-shot baseline on the held-out test set at each step, with the exception of PAAWS FL under late fusion, where the gap is less pronounced. This pattern suggests that the performance gains are stable across the incremental learning process, rather than driven by the addition of any single activity.

5.2 Visualization of classifier boundaries evolution

To illustrate how the model evolves as new activities are introduced, we visualize the learned feature space across incremental steps. While metrics such as F1 score summarize performance, they do not reveal how decision regions shift or how co-occurrence relationships are represented. Feature space visualization enables us to observe emerging clusters, co-occurring activities, and targeted updates to existing classifier heads.

For each sample, we extract per-limb embeddings from the frozen encoder, flatten them, and project them to two dimensions using UMAP [44], fitted jointly on training samples for stability. Points are colored by model predictions (i.e., which classification head was activated), with gray indicating no activation. Soft density contours show the coverage of the four seed heads.

We show the snapshots of model evolution in Figure 4. In Step 0, only the seed activities (*"walking"*, *"sitting"*, *"standing"*, *"cycling"*) are present. In Step 1, *"lying on left side"* is added, turning a previously unassigned cluster into a new region (purple samples). No existing classifiers were activated on the new samples, so no retraining or updates were triggered. In Step 2, activity *"treadmill (4mph)"* were added, the model predicted co-occurence with *"walking"*. Since the co-occurence detection was corrected (confirmed by user), no retraining or updates are

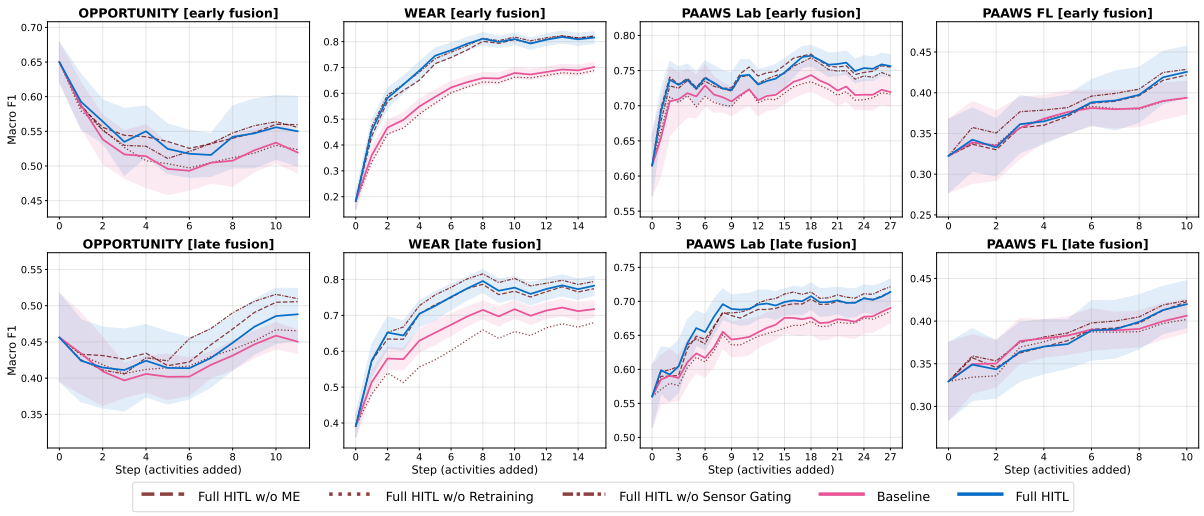


Fig. 3. Macro F1 progression over steps.

triggered. In Step 3, new activity *"chopping food"* was added, the system predicted co-occurence with *"walking"* and *"standing"*. The user confirmed *"standing"* but rejected *"walking"*, so a retraining episode is triggered on the *"walking"* classifier (we can see from the UMAP visualization, the dark blue region was shrunk after the update).

These visualizations provide geometric evidence that the system evolves in a principled manner: co-occurrence relationships are grounded in genuine feature similarity, targeted retraining updates only the affected heads, and unrelated clusters remain undisturbed as new activities are incrementally introduced.

5.3 Analysis of the impact of sensor gating on model performance

Although aggregate results (Table 3) suggest that sensor gating contributes little to overall performance, a closer per-activity analysis reveals a markedly different picture (see Figure 5). The effect of gating is highly uneven, with improvements for some activities and clear degradation for others.

The PAAWS Lab and PAAWS FL dataset share similar sensor configurations and overlapping activity sets, thus having similar patterns. Across both datasets, gating tends to benefit activities with distinct and spatially localized sensor signatures. For example, activities with isolated sensor differences (easy to gate) in PAAWS Lab such as *"Lying on right side"* (▲ + .14), *"Lying on left side"* (▲ + .08), and *"Lying on back"* (▲ + .07), as well as repetitive upper-body or structured movements in PAAWS FL such as *"Flossing teeth"* (▲ + .10), *"Sweeping"* (▲ + .04), and *"Dry mopping"* (▲ + .03), show consistent improvements. In contrast, activities that are highly similar to one another and rely on distributed, full-body motion patterns show limited gains or degradation. This is particularly evident for walking-related variations: in PAAWS Lab, activities such as different variations of *"Treadmill (3 mph)"* and *"Walk downstairs"* (▼ − .03) do not benefit from gating, and in PAAWS FL, *"Walking treadmill"* (▼ − .14), *"Walking fast"* (▼ − .006), and *"Walk downstairs"* (▼ − .02) show similar trends. Sensor gating works best when the useful signal comes from a small set of sensors, where it can reduce noise and focus the model. Sensor gating, however, less helpful for groups of similar activities that rely on signals from many sensors, where the differences are subtle and not always obvious to humans.

In the Opportunity dataset, the overall effect of sensor gating is relatively small despite the presence of additional object-mounted sensors. While modest gains are observed for activities such as *"Close fridge"* (▲ + .04)

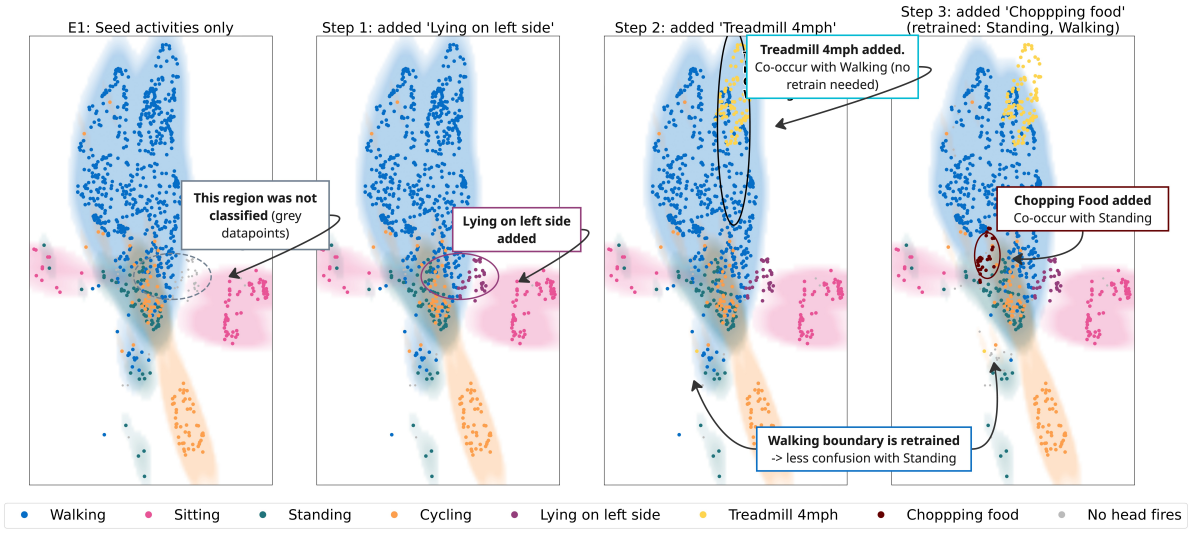


Fig. 4. Snapshot of how the activity boundary refined as new activities were added (PAAWS Lab; subject DS_11). Each point represents a test window, colored by which binary head fires on it; gray points are not claimed by any trained head. Soft shaded regions show the coverage of each seed head (Step 0). After seed training, four locomotion activities are recognized; the *"lying on left side"* cluster (dashed circle) has no head yet and remains gray. (Step 1) A *"lying on left side"* class is added — the previously gray cluster is now claimed. (Step 2) A *"treadmill 4mph"* activity is added; the *"treadmill 4mph"* cluster appears (yellow datapoints), while the *"walking"* classifier (dark blue shaded region) continues to be activated on it, illustrating co-occurrence. (Step 3) A *"chopping food"* activity is added; the *"walking"* region is updated to correct false positives detected by adding the new activity.

and *"Drink from cup"* (▲ + .03), performance decreases for others including *"Clean table"* (▼ − .07) and *"Close drawer"* (▼ − .05). Notably, activities that involve the same object but opposite actions (e.g., *"Open door"* vs. *"Close door,"* *"Open drawer"* vs. *"Close drawer"*) are difficult to separate, and improvements are often observed for only one of the paired actions. Others errors are from gestures that have no dedicated object motion sensor, such as *"Clean table"* with F1 score ▼ − .07 (no sensor on the table, so we could only mark the motion sensor in knives and plates as "high importance"), or *"Toggle switch"* (we marked all object sensors as "low importance" for this activity).

In the WEAR dataset, the impact of sensor gating is even more limited, with only minor improvements for activities such as *"Sit ups complex"* (▲ + .03) and *"Lunges complex"* (▲ + .03), and small decreases for activities like *"Jogging skipping"* (▼ − .05). Most activities in WEAR involve full-body movement, making it difficult—even for domain experts—to identify a small subset of sensors that are most informative. The dataset also contains many groups of closely related activities (e.g., variations of jogging and lunging) with highly similar motion patterns, where restricting the model to a subset of sensors removes subtle but important differences needed for discrimination.

Together, these results suggest that sensor gating is highly activity-dependent. It tends to help when the discriminative signal is spatially localized (e.g., in postures or repetitive movements) where focusing on a subset of sensors reduces noise and improves separability. However, it can hurt when recognition depends on distributed, whole-body patterns, because hard gating may remove complementary signals needed to capture coordinated motion. Importantly, sensors that appear "irrelevant" for a given activity may still provide useful negative

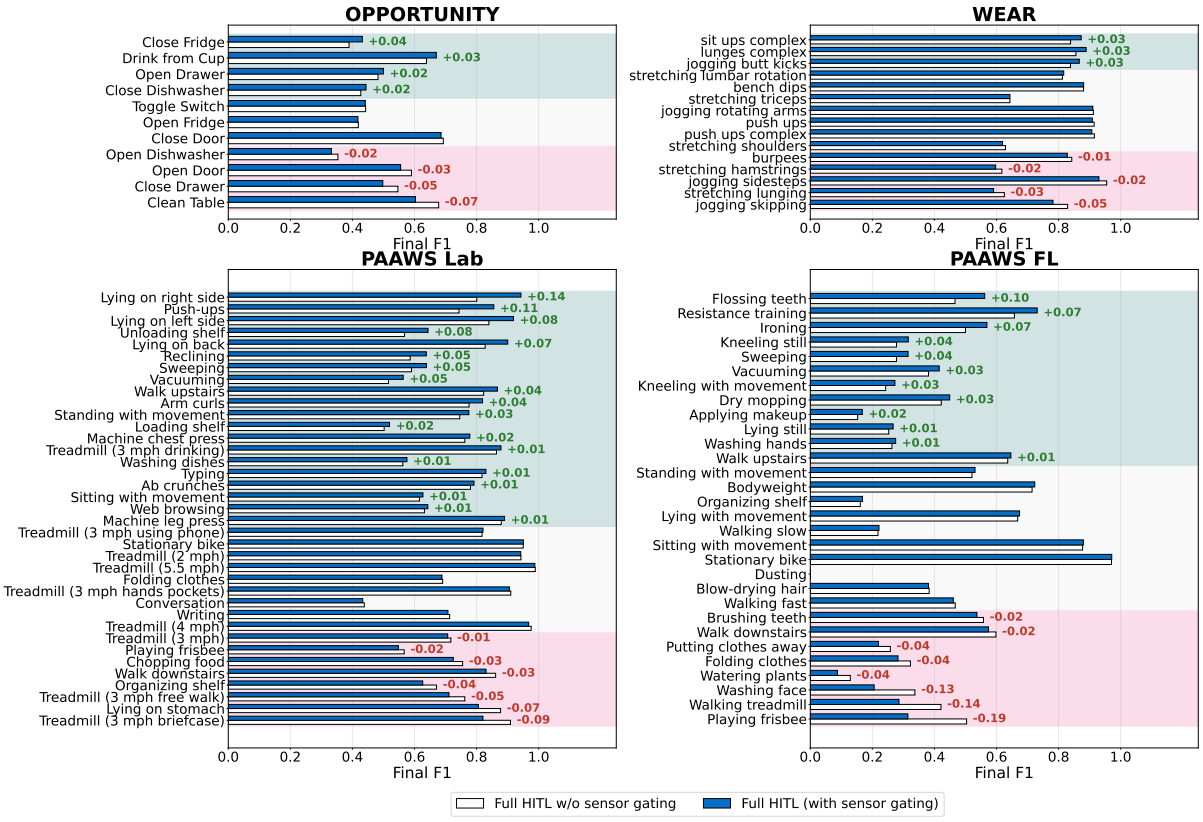


Fig. 5. Impact of sensor gating on the performance of the model.

evidence: for example, the absence of motion in certain body locations can help distinguish between similar activities. By suppressing these signals, gating can reduce the model's ability to differentiate between classes.

5.4 Analysis of the impact of ME constraints on model performance

Aggregate results (Table 3) suggest that mutual exclusion (ME) constraints have limited overall impact, in this section, we look closer at the impact of ME constraint on model performance (Figure 6).

The strongest and most consistent degradation is observed in the Opportunity dataset. While a few activities benefit, such as *"Close drawer"* (▲ + .07) and *"Close door"* (▲ + .06), many others show clear drops, including *"Open dishwasher"* (▼ − .09), *"Toggle switch"* (▼ − .06), and *"Drink from cup"* (▼ − .05). Some of these errors can be attributed to the presence of paired actions on the same object (e.g., *"Open drawer"* vs. *"Close drawer"*, *"Open door"* vs. *"Close door"*), where enforcing mutual exclusion can suppress the correct prediction when the model confuses the pair.

In contrast, the other datasets show more mixed and generally smaller effects. In the PAAWS Lab and PAAWS FL datasets, ME constraints lead to both improvements and degradations across activities, while in the WEAR dataset the overall impact remains modest. These results suggest that the benefit of ME constraints depends heavily on how well the defined exclusions align with true activity relationships and can be harmful when applied to closely related activity pairs.

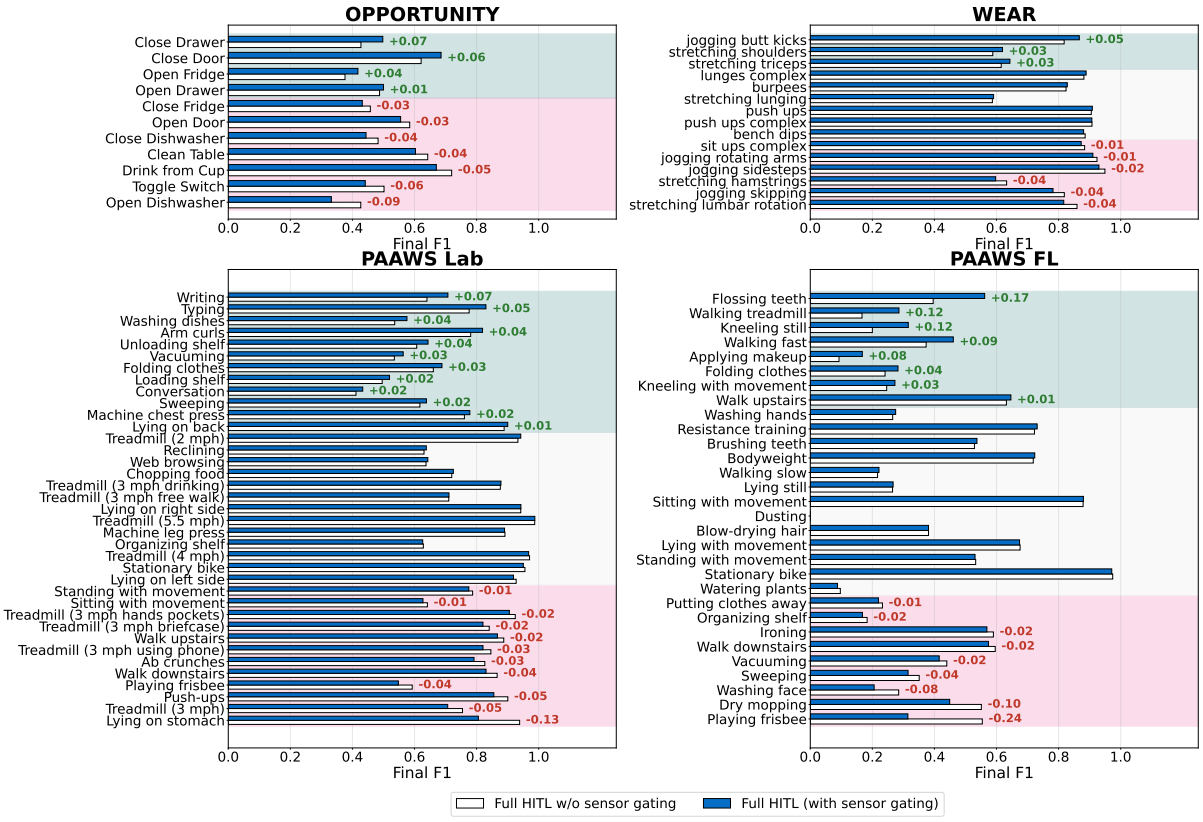


Fig. 6. Impact of ME constraints on the performance of the model.

5.5 Analysis of the impact of seed activity choice on model evolution

We analyze how the choice of seed activities affects model evolution by examining the number of retraining events triggered for each class (Figure 7). Across the PAAWS Lab, PAAWS FL, and WEAR datasets, most retraining events occur on the seed activities rather than the newly added ones. For example, in the PAAWS Lab dataset (20 participants), *"Standing"* and *"Sitting"* are retrained over 100 times across 20 participants, and new activities account for 121 retrains in total; in the PAAWS FL dataset (20 participants), seed activities such as *"Walking"* (73 retrain episodes) and *"Standing"* (68 retrain episodes) dominate retraining, compared to only 14 retrain episodes across all new activities. A similar pattern appears in the WEAR dataset (23 participants), where seed activities such as *"Lunges"* (86 retrain episodes), *"Sit ups"* (78 retrain episodes), and *"Jogging"* (70 retrain episodes) receive more retraining than most added activities. This suggests that in these datasets, newly added activities are often closely related to the seed activities and primarily serve to refine their decision boundaries.

In contrast, results on the Opportunity dataset (4 participants) shows a different pattern, where a larger share of retraining and improvement in F1 scores occurs among newly added activities. Here, seed activities are posture-based (e.g., *"Sit," "Stand," "Lie"*), and added activities are object-centric actions (e.g., interacting with doors or drawers) that depend on different sensors. Many of these activities appear in paired forms (e.g., *"Open door"* vs. *"Close door,"*) so adding one often triggers updates in its paired counterpart rather than the seed activities. The

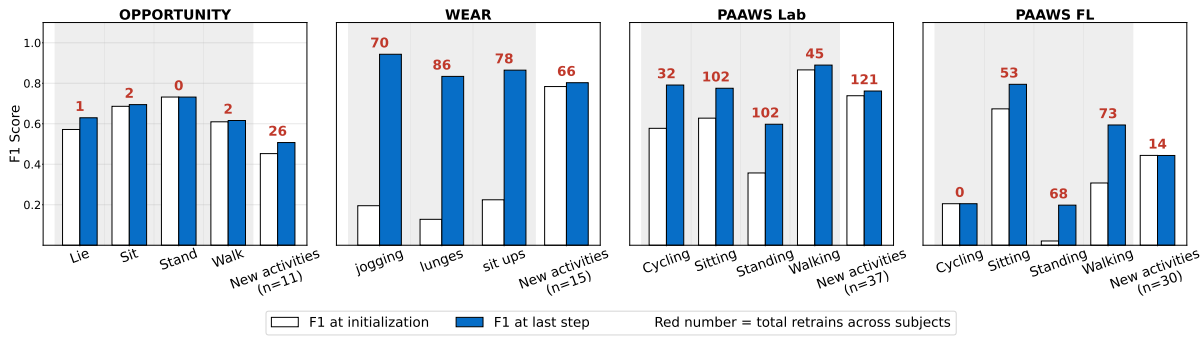


Fig. 7. Impact of seed-activity choice on incremental learning. Macro F1 at initialization vs. final step and the number of retrain episodes across all participants.

choice of seed activities has an influence the evolution of the model (in both F1 score performance and number of retrain episode).

6 Discussion and Future Work

In this section, we discuss the implications of our work for the human-in-the-loop activity recognition system and provide suggestions for combining it with existing techniques in the literature to enable real-time, lifelong activity recognition in the wild.

6.1 How human-in-the-loop interactions could improve HAR model prediction

Recent advances in HAR have largely focused on developing increasingly powerful, generalized models, often trained on large-scale datasets with the expectation that they will generalize across users and contexts [18, 53, 80]. This generalization, however, has been challenging because datasets vary in their activities, making it difficult to robustly deploy and evaluate these HAR models in the real world, where the set of activities can change. Human-in-the-loop systems have the potential to improve the generalization of HAR models to new scenarios. These systems, however, have not been much explored. Existing systems that incorporate user feedback typically rely on lightweight interactions, such as in-the-moment self-reports or confirmation of predicted activities [3, 24]. Some approaches also allow users to inject domain knowledge into structured or knowledge-based models [26]. While these methods demonstrate the feasibility of integrating user input, they primarily treat users as sources of labels or validation signals [50], rather than as active collaborators in tuning model behavior.

In this work, we present a framework that incorporates multiple forms of user feedback (co-occurrence, mutual exclusiveness, and sensor relevance) to support model training and adaptation in a few-shot setting [62]. Our results show that detecting co-occurring activities and performing targeted retraining during the addition of new activities can improve model performance over time, particularly in refining existing activity boundaries. We also explore incorporating additional forms of domain knowledge, such as sensor importance and mutual exclusivity between activities. While the effectiveness of these signals varies depending on the activity and sensing configuration, our framework demonstrates the feasibility of integrating richer forms of user knowledge into the learning process.

Our findings demonstrate the promise and the broader potential of enabling multiple forms of user-guided knowledge injection into HAR models. Future work should explore a wider range of human-in-the-loop interactions, potentially tailored to specific sensing setups and application contexts. For example, users could remove irrelevant activity classes, provide contextual constraints such as location-based activation of classifiers [76], or

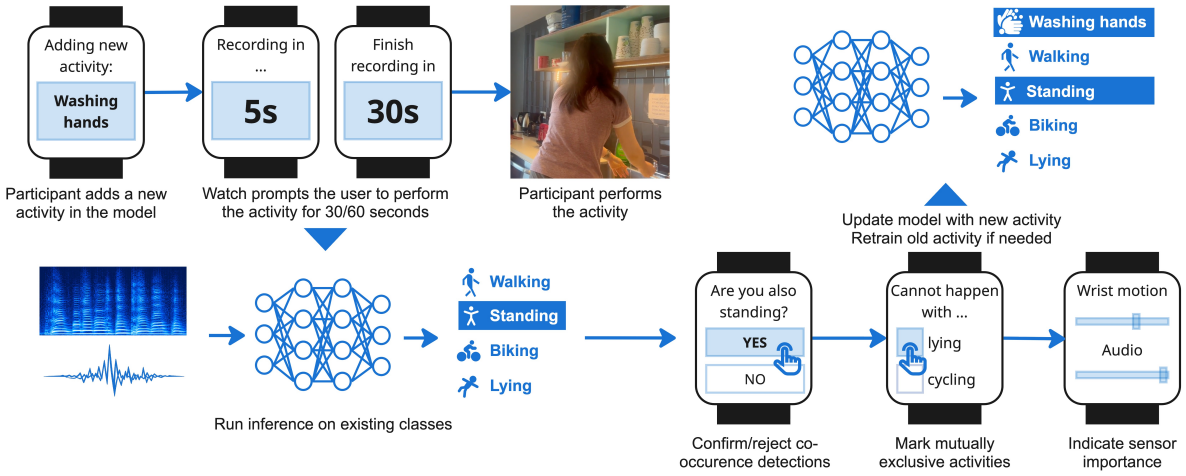


Fig. 8. How our framework could be integrated with micro-interactions on the smartwatch

adjust sensor usage to balance performance and privacy [38]. Exploring these interaction mechanisms remains an important direction, and future work could use our framework as a starting point and add relevant novel interactions.

6.2 Integration with microinteraction EMA (μEMA) for real-time systems

Prior work has explored a wide range of methods for collecting activity labels, including end-of-day recall and reflection-based logging [25, 35, 63, 66]. More recently, researchers have introduced speech-based and multimodal interactions on smartwatches to support lightweight, in-situ data collection [2, 27]. In many cases, however, these collected labels are primarily used for model evaluation or validation rather than for continuous training and adaptation in real time [50].

Our proposed framework presents an opportunity to move beyond passive data collection toward lightweight, human-in-the-loop activity recognition systems that operate directly on wearable devices. In particular, integrating the framework with micro-EMA (μEMA) is promising, as prior work has shown that such micro-interactions can be sustained over long periods with low user burden [51]. This makes μEMA well-suited for studying how models adapt to evolving activities, contexts, and routines in longitudinal, real-world settings.

We envision some interaction mechanisms enabled by this integration. First, users could define new activities through few-shot interactions (see Figure 8): performing an activity for a short duration (e.g., 30–60 seconds) to provide initial training data, allowing the system to quickly instantiate new classifiers [37, 73]. Second, the system can passively detect potential novel activities, e.g., when none of the existing classifiers are activated with enough confidence, and selectively query the user for clarification. These queried segments could then be incorporated as training samples, reducing the need for users to proactively provide samples for all activities of interest.

While our frameworks enable many such interesting amd novel interactions. There are many open challenges in real-time, free-living deployments that future work should explore. Some challenges include assessing,ecological validity, long-term user engagement, limited screen space in wearables, and the trade-offs between model performance, latency, battery life, and user burden in naturalistic settings [77].

6.3 Towards human-centered, customizable, interpretable activity recognition systems

This work leads an initial step toward human-centered, user-customizable HAR systems. Such systems enable individuals to track activities that are personally meaningful, rather than being constrained to a fixed, predefined label set [69, 70]. More broadly, they provide a flexible platform for researchers to rapidly develop activity classifiers tailored to specialized study populations or health outcomes, particularly in cases where existing datasets are limited or unrepresentative (e.g., wheelchair users or niche behaviors) [7, 21]. Supporting adaptation to new activities, as well as evolving sensor configurations, opens up new opportunities for personalized tracking, interactive systems, and context-aware interventions.

At the same time, our results highlight important challenges in designing effective human-in-the-loop systems. In some cases, user interactions can negatively impact model performance, underscoring the need for systems that can communicate effectively with users. Future HAR systems should be designed to (1) convey their limitations, such as when certain activities cannot be reliably learned [73], (2) support debugging when performance degrades (e.g., due to distribution shift, overfitting, or sensor failure) [26], and (3) enable users to take actions to correct the model [71]. Achieving this through lightweight interactions remains challenging, particularly given the screen real-estate and input options on wearable and mobile devices [68, 78].

Recent advances in large language models (LLMs) and interpretable machine learning offer promising directions to address these challenges [5, 9]. Emerging work suggests that LLMs can help translate model behavior and sensor data into human-understandable explanations, enabling more effective communication of system limitations and supporting interactive debugging [12, 22, 33, 67, 80]. Integrating such capabilities into HAR systems may allow users to better understand, diagnose, and refine model behavior through natural, low-effort interactions.

7 Limitations

This work has several limitations. First, our evaluation is conducted offline using existing datasets, necessitating the simulation of adding new activities. While this enables controlled experimentation, real-world settings are substantially more complex. In free-living, activities may overlap, transitions may be ambiguous, and some activities may not be distinguishable given the available signals. Moreover, our setup does not allow for longitudinal evaluation, limiting our ability to study how the system performs under sustained use, evolving routines, and long-term adaptation.

Second, the human-in-the-loop interactions in our study are simulated and performed by researchers. As domain experts, they likely provide cleaner and more consistent feedback than typical users, representing a best-case scenario. In real-world deployments, user interactions are expected to be noisier, less consistent, and influenced by varying levels of understanding of the system, introducing error into labeling, which may affect both model performance and usability.

Third, our experiments are limited to accelerometer-based sensing from objects and body-worn devices. In real-world applications, HAR systems often incorporate a wider range of sensing modalities, such as audio, vision, physiological signals, and location. These modalities may provide more salient or complementary signals for certain activities and could make it easier for users to provide meaningful feedback (e.g., associating *"walking"* with motion or *"talking"* with audio). Exploring how the proposed framework extends to multimodal sensing environments remains an important direction for future work.

Finally, our implementation relies on a single representation learning backbone (SimCLR) [55]. While the proposed framework is model-agnostic and can be applied to other architectures, we do not evaluate its performance across different backbones. As a result, the extent to which our findings generalize to alternative representation learning methods remains an open question.

8 Conclusion

We introduce a human-in-the-loop incremental learning framework for activity recognition that enables adaptive, multi-label modeling with minimal data. Our results demonstrate the potential of integrating user feedback to guide model updates, improving robustness and addressing key limitations of few-shot learning in real-world settings. More broadly, this work highlights a shift toward human-centered HAR systems, where users could actively shape model behavior rather than just passively providing data. Our work has several implications for future research on interactive, interpretable, and continuously adaptive sensing systems for real-world deployment.

Acknowledgements

This research is partially supported by the National Institutes of Health, under award number NIDA P30DA029926, and the National Science Foundation, under award number IIS-2442593, and ... The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the sponsors. Any mention of specific companies or products does not imply any endorsement by the authors, by their employers, or by the sponsors.

References

[1] Rebecca Adaimi and Edison Thomaz. Lifelong adaptive machine learning for sensor-based human activity recognition using prototypical networks. *Sensors*, 22(18):6881, September 2022.

[2] Riku Arakawa, Jill Fain Lehman, and Mayank Goel. PrISM-Q&A: Step-aware voice assistant on a smartwatch enabled by multimodal procedure tracking and large language models. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 8(4):1–26, November 2024.

[3] Riku Arakawa, Hiromu Yakura, Vimal Mollyn, Suzanne Nie, Emma Russell, Dustin P. DeMeo, Haarika A. Reddy, Alexander K. Maytin, Bryan T. Carroll, Jill Fain Lehman, and Mayank Goel. PrISM-Tracker: A framework for multimodal procedure tracking using wearable sensors and state transition information with user-driven handling of errors and uncertainty. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 6(4):1–27, December 2022.

[4] Ling Bao and Stephen S. Intille. Activity recognition from user-annotated acceleration data. In Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell, Oscar Nierstrasz, C. Pandu Rangan, Bernhard Steffen, Demetri Terzopoulos, Dough Tygar, Moshe Y. Vardi, Alois Ferscha, and Friedemann Mattern, editors, *Pervasive Computing*, volume 3001, pages 1–17. Springer Berlin Heidelberg, 2004.

[5] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, and Antonio Torralba. Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences*, 117(48):30071–30078, December 2020.

[6] Marius Bock, Hilde Kuehne, Kristof Van Laerhoven, and Michael Moeller. WEAR: An outdoor sports dataset for wearable and egocentric activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 8(4):1–21, November 2024.

[7] Rachel L Carey, Ha Le, Donna L Coffman, Inbal Nahum-Shani, Mohanraj Thirumalai, Cole Hagen, Laura A Baehr, Mary Schmidt-Read, Marlyn S R Lamboy, Stephanie A Kolakowsky-Hayner, Ralph J Marino, Stephen S Intille, and Shivayogi V Hiremath. mHealth-based just-in-time adaptive intervention to Improve the physical activity levels of individuals with spinal cord injury: Protocol for a randomized controlled trial. *JMIR Research Protocols*, 13:e57699, June 2024.

[8] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A Simple Framework for Contrastive Learning of Visual Representations. *International Conference on Machine Learning*, pages 1597–1607, 2020.

[9] Wenqiang Chen, Jiaxuan Cheng, Leyao Wang, Wei Zhao, and Wojciech Matusik. Sensor2Text: Enabling natural language interactions for daily activity tracking using wearable sensors. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 8(4):1–26, November 2024.

[10] Heng-Tze Cheng, Feng-Tso Sun, Martin Griss, Paul Davis, Jianguo Li, and Di You. NuActiv: Recognizing unseen new activities using semantic attribute-based learning. In *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services*, pages 361–374. ACM, June 2013.

[11] Woohyeok Choi, Sangkeun Park, Duyeon Kim, Youn-kyung Lim, and Uichin Lee. Multi-stage receptivity model for mobile just-in-time health Intervention. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(2):1–26, June 2019.

[12] Akshat Choube, Ha Le, Jiachen Li, Kaixin Ji, Vedant Das Swain, and Varun Mishra. GLOSS: Group of LLMs for open-ended sensemaking of passive sensing data for health and wellbeing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 9(3):30, 2025.

[13] Mathias Ciliberto, Vitor Fortes Rey, Alberto Calatroni, Paul Lukowicz, and Daniel Roggen. Opportunity++: A multimodal dataset for video- and wearable, object and ambient sensors-based human activity recognition. *Frontiers in Computer Science*, 3:792065, December 2021.

[14] Dagoberto Cruz-Sandoval, Jessica Beltran-Marquez, Matias Garcia-Constantino, Luis A. Gonzalez-Jasso, Jesus Favela, Irvin Hussein Lopez-Nava, Ian Cleland, Andrew Ennis, Netzahualcoyotl Hernandez-Cruz, Joseph Rafferty, Jonathan Synnott, and Chris Nugent. Semi-automated data labeling for activity recognition in pervasive healthcare. *Sensors*, 19(14):3035, July 2019.

[15] Sean T. Doherty, Christopher J. Lemieux, and Culum Canally. Tracking human activity and well-being in natural environments using wearable sensors and experience sampling. *Social Science & Medicine*, 106:83–92, April 2014.

[16] David H. Epstein, Matthew Tyburski, William J. Kowalczyk, Albert J. Burgess-Hull, Karran A. Phillips, Brenda L. Curtis, and Kenzie L. Preston. Prediction of stress and drug craving ninety minutes in the future with passively collected GPS data. *npj Digital Medicine*, 3(1):26, December 2020.

[17] Harish Haresamudram, Apoorva Beedu, Varun Agrawal, Patrick L. Grady, Irfan Essa, Judy Hoffman, and Thomas Plötz. Masked reconstruction based self-supervision for human activity recognition. In *Proceedings of the 2020 International Symposium on Wearable Computers*, pages 45–49. ACM, September 2020.

[18] Harish Haresamudram, Chi Ian Tang, Sungho Suh, Paul Lukowicz, and Thomas Plötz. Past, present, and future of sensor-based human activity recognition using wearables: A surveying tutorial on a still challenging task. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 9(2):1–44, June 2025.

[19] Joyce Ho and Stephen S. Intille. Using context-aware computing to reduce the perceived burden of interruptions from mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 909–918. ACM, April 2005.

[20] Alexander Hoelzemann and Kristof Van Laerhoven. A matter of annotation: An empirical study on in situ and self-recall activity annotations from wearable sensors. *Frontiers in Computer Science*, 6:1379788, July 2024.

[21] Sungjin Hwang, Zikang Leng, Seungwoo Oh, Kwanguk Kim, and Thomas Plötz. More data for people with disabilities! Comparing data collection efforts for wheelchair transportation mode detection. In *Proceedings of the 2024 ACM International Symposium on Wearable Computers*, pages 82–88. ACM, October 2024.

[22] Sheikh Asif Imran, Mohammad Nur Hossain Khan, Subrata Biswas, and Bashima Islam. LLaSA: A sensor-aware LLM for natural language reasoning of human activity from IMU data, September 2025. arXiv:2406.14498 [cs].

[23] Stephen Intille, Caitlin Haynes, Dharam Maniar, Aditya Ponnada, and Justin Manjourides. micro-EMA: Microinteraction-based ecological momentary assessment (EMA) using a smartwatch. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 1124–1128. ACM, September 2016.

[24] Amir Khan, Zunaira Jamil, Muhammad Daud Abdullah Asif, Shah Khalid, Yusra Arshad, and Rizwan Ahmad. A human-in-the-loop class-incremental framework for lifelong human activity recognition. In *2025 27th International Multitopic Conference (INMIC)*, pages 1–6. IEEE, December 2025.

[25] Hossein Khayami, Lining Wang, Young-Ho Kim, Bongshin Lee, David E. Conroy, Amanda Lazar, Eun Kyoung Choe, and Hernisa Kacorri. From verbal reports to personalized activity trackers: Understanding the challenges of ground truth data collection with older adults in the wild. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 9(2):1–33, June 2025.

[26] Mohammad Kianpisheh, Alex Mariakakis, and Khai N. Truong. exHAR: An interface for helping non-experts develop and debug knowledge-based human activity recognition systems. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 8(1):1–30, March 2024.

[27] Young-Ho Kim, Diana Chou, Bongshin Lee, Margaret Danilovich, Amanda Lazar, David E. Conroy, Hernisa Kacorri, and Eun Kyoung Choe. MyMove: Facilitating older adults to collect in-situ activity labels on a smartwatch with speech. In *CHI Conference on Human Factors in Computing Systems*, pages 1–21, New York, NY, USA, April 2022. ACM.

[28] Young-Ho Kim, Jae Ho Jeon, Bongshin Lee, Eun Kyoung Choe, and Jinwook Seo. OmniTrack: A flexible self-tracking approach leveraging semi-automated tracking. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3):1–28, September 2017.

[29] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

[30] Joseph Korpela, Takayuki Akiyama, Takehiro Niikura, and Katsuyuki Nakamura. Reducing label fragmentation during time-series data annotation to reduce annotation costs. In *Adjunct Proceedings of the 2021 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2021 ACM International Symposium on Wearable Computers*, pages 328–333, Virtual USA, September 2021. ACM.

[31] Jennifer R. Kwapisz, Gary M. Weiss, and Samuel A. Moore. Activity recognition using cell phone accelerometers. *ACM SIGKDD Explorations Newsletter*, 12(2):74–82, March 2011.

[32] Hyeokhyen Kwon, Gregory D. Abowd, and Thomas Plötz. Handling annotation uncertainty in human activity recognition. In *Proceedings of the 23rd International Symposium on Wearable Computers*, pages 109–117. ACM, September 2019.

[33] Ha Le, Akshat Choube, Varun Mishra, and Stephen Intille. Feasibility of using a multi-agent LLM system to correct annotations and support low-effort activity labeling. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, page 36, 2026.

Under Review.

[34] Ha Le, Rithika Lakshminarayanan, Jixin Li, Varun Mishra, and Stephen Intille. Collecting self-reported physical activity and posture data using audio-based ecological momentary assessment. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 8(3):1–35, 2024.

[35] Ha Le, Veronika Potter, Akshat Choube, Rithika Lakshminarayanan, Varun Mishra, and Stephen Intille. A context-assisted, semi-automated activity recall interface allowing uncertainty. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 9(4):33, 2025.

[36] Ha Le, Veronika Potter, Rithika Lakshminarayanan, Varun Mishra, and Stephen Intille. Feasibility and utility of multimodal micro ecological momentary assessment on a smartwatch. *CHI Conference on Human Factors in Computing Systems (CHI' 25)*, 2025.

[37] Ying Lei, Yancheng Cao, Will Ke Wang, Yuanzhe Dong, Changchang Yin, Weidan Cao, Ping Zhang, Jingzhe Yang, Bingsheng Yao, Yifan Peng, Chunhua Weng, Randy Auerbach, Lena Mamykina, Dakuo Wang, Yuntao Wang, and Xuhai Xu. WatchGuardian: Enabling user-defined personalized just-in-time intervention on smartwatch. *ACM Transactions on Computing for Healthcare*, page 3788689, January 2026.

[38] Jiachen Li, Bingrui Zong, Tingyu Cheng, Yunzhi Li, Elizabeth D. Mynatt, and Ashutosh Dhekne. Privacy vs. awareness: Relieving the tension between older adults and adult children when sharing in-home activity data. *Proceedings of the ACM on Human-Computer Interaction*, 7(CSCW2):1–30, September 2023.

[39] Guoliang Lin, Yongheng Xu, Hanjiang Lai, and Jian Yin. Revisiting few-shot learning from a causal perspective. *IEEE Transactions on Knowledge and Data Engineering*, 36(11):6908–6919, November 2024.

[40] Mengxi Liu, Sizhen Bian, Bo Zhou, and Paul Lukowicz. iKAN: Global incremental learning with KAN for human activity recognition across heterogeneous datasets. In *Proceedings of the 2024 ACM International Symposium on Wearable Computers*, pages 89–95. ACM, October 2024.

[41] T.-Y. Liu. A context-aware ubiquitous learning environment for language listening and speaking. *Journal of Computer Assisted Learning*, 25(6):515–527, December 2009.

[42] Charles E. Matthews. Twenty-four hour physical activity recall (24PAR) system interviewer training materials/protocol.

[43] Alan Mazankiewicz, Klemens Böhm, and Mario Berges. Incremental real-time personalization in human activity recognition using domain adaptive batch normalization. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 4(4):1–20, December 2020.

[44] Leland McInnes, John Healy, and James Melville. Uniform manifold approximation and projection. *Nature Reviews Methods Primers*, 4(1):83, November 2024.

[45] Shenghuan Miao, Ling Chen, and Rong Hu. Spatial-temporal masked autoencoder for multi-device wearable human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 7(4):1–25, December 2023.

[46] Varun Mishra, Byron Lowens, Sarah Lord, Kelly Caine, and David Kotz. Investigating contextual cues as indicators for EMA delivery. In *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*, pages 935–940. ACM, September 2017.

[47] Subigya Nepal, Wenjun Liu, Arvind Pillai, Weichen Wang, Vlado Vojdanovski, Jeremy F. Huckins, Courtney Rogers, Meghan L. Meyer, and Andrew T. Campbell. Capturing the college experience: A four-year mobile sensing study of mental health, resilience and behavior of college students during the pandemic. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 8(1):1–37, March 2024.

[48] Stavros Ntalampiras and Manuel Roveri. An incremental learning mechanism for human activity recognition. In *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–6, Athens, Greece, December 2016. IEEE.

[49] Aditya Ponnada, Jixin Li, Shirlene Wang, Wei-Lin Wang, Bridgette Do, Genevieve F. Dunton, and Stephen S. Intille. Contextual biases in microinteraction ecological momentary assessment (micro-EMA) non-response. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 6(1):1–24, March 2022.

[50] Aditya Ponnada, Binod Thapa-Chhetry, Justin Manjourides, and Stephen Intille. Measuring criterion validity of microinteraction ecological momentary assessment (Micro-EMA): Exploratory pilot Study with physical activity measurement. *JMIR mHealth and uHealth*, 9(3):e23391, March 2021.

[51] Aditya Ponnada, Shirlene Wang, Daniel Chu, Bridgette Do, Genevieve Dunton, and Stephen Intille. Intensive longitudinal data collection using microinteraction ecological momentary assessment: Pilot and preliminary results. *JMIR Formative Research*, 6(2):e32772, February 2022.

[52] Veronika Potter, Hoan Tran, Daniel Mobley, Suzanne M. Bertisch, Dinesh John, and Stephen Intille. The Physical Activity Assessment Using Wearable Sensors (PAAWS) Dataset: Labeled Laboratory and Free-Living Accelerometer Data. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 9(4):1–32, December 2025.

[53] Minghui Qiu, Cekai Weng, Mingming Fan, and Kaishun Wu. Towards customizable foundation models for human activity recognition with wearable devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 9(3):1–29, September 2025.

[54] Attila Reiss. PAMAP2 physical activity monitoring, 2012.

[55] Aaqib Saeed, Ye Li, Tanir Ozcelebi, and Johan Lukkien. Multi-sensor data augmentation for robust sensing. In *2020 International Conference on Omni-layer Intelligent Systems (COINS)*, pages 1–7, Barcelona, Spain, August 2020. IEEE.

[56] Philip Schmidt, Attila Reiss, Robert Dürichen, Claus Marberger, and Kristof Van Laerhoven. Introducing WESAD, a multimodal dataset for wearable stress and affect detection.

[57] Guangyuan Shi, Jiaxin Chen, Wenlong Zhang, Li-Min Zhan, and Xiao-Ming Wu. Overcoming catastrophic forgetting in incremental few-shot learning by finding flat minima. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, pages 6747 – 6761. Curran Associates Inc., 2021.

[58] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. *Advances in neural information processing systems*, (30):10, 2017.

[59] Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. Incremental learning of object detectors without catastrophic forgetting. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3420–3429. IEEE, October 2017.

[60] Joshua M. Smyth and Author A. Stone. Ecological momentary assessment research in behavioral medicine. *Journal of Happiness Studies*, 4(1):35–52, 2003.

[61] Marija Stojchevska, Mathias De Brouwer, Martijn Courteaux, Femke Ongenae, and Sofie Van Hoecke. From lab to real world: Assessing the effectiveness of human activity recognition and optimization through personalization. *Sensors*, 23(10):4606, May 2023.

[62] Megha Thukral, Harish Haresamudram, and Thomas Plötz. Cross-domain HAR: Few-shot transfer learning for human activity recognition. *ACM Transactions on Intelligent Systems and Technology*, 16(1):1–35, February 2025.

[63] Ye Tian, Xiaoyuan Ren, Zihao Wang, Onat Gungor, Xiaofan Yu, and Tajana Rosing. DailyLLM: Context-aware activity log generation using multi-modal sensors and LLMs, July 2025.

[64] Yonatan Vaizman, Katherine Ellis, and Gert Lanckriet. Recognizing detailed human context in the wild from smartphones and smartwatches. *IEEE Pervasive Computing*, 16(4):62–74, October 2017.

[65] Yonatan Vaizman, Katherine Ellis, Gert Lanckriet, and Nadir Weibel. ExtraSensory App: Data collection in-the-wild with rich user interface to self-report behavior. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pages 1–12. ACM, April 2018.

[66] Yonatan Vaizman, Nadir Weibel, and Gert Lanckriet. Context recognition in-the-wild: Unified model for multi-modal sensors and multi-label classification. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(4):1–22, January 2018.

[67] Chongyang Wang, Yuan Feng, Lingxiao Zhong, Siyi Zhu, Chi Zhang, Siqi Zheng, Chen Liang, Yuntao Wang, Chengqi He, Chun Yu, and Yuanchun Shi. UbiPhysio: Support daily functioning, fitness, and rehabilitation with action understanding and feedback in natural language. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 8(1):1–27, March 2024.

[68] Xinru Wang, Mengjie Yu, Hannah Nguyen, Michael Iuzzolino, Tianyi Wang, Peiqi Tang, Natasha Lynova, Co Tran, Ting Zhang, Naveen Sendhilnathan, Hrvoje Benko, Haijun Xia, and Tanya Jonker. Less or more: Towards glanceable explanations for LLM recommendations using ultra-small devices, February 2025.

[69] Yiwen Wang, Hossein Khayami, Bongshin Lee, Amanda Lazar, Hernisa Kacorri, and Eun Kyoung Choe. Enabling older adults to provide high-quality activity labels: Unpacking accuracy, precision, and granularity in activity labeling. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 9(4):1–24, December 2025.

[70] Yiwen Wang, Mengying Li, Young-Ho Kim, Bongshin Lee, Margaret Danilovich, Amanda Lazar, David E Conroy, Hernisa Kacorri, and Eun Kyoung Choe. Redefining activity tracking through older adults' reflections on meaningful activities. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 1–15. ACM, May 2024.

[71] Zijie J. Wang, Alex Kale, Harsha Nori, Peter Stella, Mark E. Nunnally, Duen Horng Chau, Mihaela Vorvoreanu, Jennifer Wortman Vaughan, and Rich Caruana. Interpretability, then what? Editing machine learning models to reflect human knowledge and values. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4132–4142. ACM, August 2022.

[72] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 374–382. IEEE, June 2019.

[73] Xuhai Xu, Jun Gong, Carolina Brum, Lilian Liang, Bongsoo Suh, Kumar Gupta, Yash Agarwal, Laurence Lindsey, Runchang Kang, Behrooz Shahsavari, Tu Nguyen, Heriberto Nieto, Scott E. Hudson, Charlie Maalouf, Seyed Mousavi, and Gierad Laput. Enabling hand gesture customization on wrist-worn devices. In *CHI Conference on Human Factors in Computing Systems*, pages 1–19, April 2022. arXiv:2203.15239 [cs].

[74] Xuhai Xu, Xin Liu, Han Zhang, Weichen Wang, Subigya Nepal, Yasaman Sefidgar, Woosuk Seo, Kevin S. Kuehn, Jeremy F. Huckins, Margaret E. Morris, Paula S. Nurius, Eve A. Riskin, Shwetak Patel, Tim Althoff, Andrew Campbell, Anind K. Dey, and Jennifer Mankoff. GLOBEM: Cross-dataset generalization of longitudinal human behavior modeling. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 6(4):1–34, December 2022.

[75] Hong Xuan, Abby Stylianou, Xiaotong Liu, and Robert Pless. Hard negative examples are hard, but useful. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, volume 12359, pages 126–142. Springer International Publishing, Cham, 2020. Series Title: Lecture Notes in Computer Science.

[76] Kiichiro Yamano and Katunobu Itou. Browsing audio life-log data using acoustic and location information. In *2009 Third International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*, pages 96–101, Sliema, Malta, October 2009. IEEE.

[77] Xinghui Yan, Yuxuan Li, Bingjian Huang, Sun Young Park, and Mark Newman. User burden of microinteractions: An in-lab experiment examining user performance and perceived burden related to in-situ self-reporting. In *Proceedings of the 23rd International Conference on Mobile Human-Computer Interaction*, pages 1–14. ACM, September 2021.

[78] Xinghui Yan, Shriti Raj, Bingjian Huang, Sun Young Park, and Mark W. Newman. Toward lightweight in-situ self-reporting: An exploratory study of alternative smartwatch interface designs in context. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 4(4):1–22, December 2020.

[79] Xiao Zhang, Hongzheng Yu, Yang Yang, Jingjing Gu, Yujun Li, Fuzhen Zhuang, Dongxiao Yu, and Zhaochun Ren. HarMI: Human activity recognition via multi-modality incremental learning. *IEEE Journal of Biomedical and Health Informatics*, 26(3):939–951, March 2022.

[80] Yuwei Zhang, Kumar Ayush, Siyuan Qiao, A. Ali Heydari, Girish Narayanswamy, Maxwell A. Xu, Ahmed A. Metwally, Shawn Xu, Jake Garrison, Xuhai Xu, Tim Althoff, Yun Liu, Pushmeet Kohli, Jiening Zhan, Mark Malhotra, Shwetak Patel, Cecilia Mascolo, Xin Liu, Daniel McDuff, and Yuzhe Yang. SensorLM: Learning the Language of Wearable Sensors, 2025.

[81] Da-Wei Zhou, Qi-Wei Wang, Zhi-Hong Qi, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. Class-incremental learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):9851–9873, December 2024.