Context Abstraction to Improve Decentralized Machine Learning in Structured Sensing Environments

Massinissa Hamidi and Aomar Osmani (✉)

LIPN-UMR CNRS 7030, Université Sorbonne Paris Nord
{hamidi,ao}@lipn.univ-paris13.fr

Abstract. In Internet of Things applications, data generated from devices with different characteristics and located at different positions are embedded into different contexts. This poses major challenges for decentralized machine learning as the data distribution across these devices and locations requires consideration for the invariants that characterize them, e.g., in activity recognition applications, the acceleration recorded by hand device must be corrected by the invariant related to the movement of the hand relative to the body. In this article, we propose a new approach that abstracts the exact context surrounding data generators and improves the reconciliation process for decentralized machine learning. Local learners are trained to decompose the learned representations into (i) universal components shared among devices and locations and (ii) local components that capture the specific context of device and location dependencies. The explicit representation of the relative geometry of devices through the special Euclidean Group $SE(3)$ imposes additional constraints that improve the decomposition process. Comprehensive experimental evaluations are carried out on sensor-based activity recognition datasets featuring multi-location and multi-device data collected in a structured sensing environment. Obtained results show the superiority of the proposed method compared with the advanced solutions.

Keywords: Meta-learning · Federated learning · Internet of things.

1 Introduction

In Internet of Things (IoT) applications, data generated from different devices (or sensors) and locations are embodied with varying contexts. The devices offer specific perspectives on the problem of interest depending on their location. The movements of the area on which the devices are positioned generate data of two different but complementary natures. For instance, in Fig. 1, the data of the movement collected from the hand sensors combines data of the whole body intertwined with data related to the movement of the hand in relation to the body. In the case of human activity recognition (HAR), we notice, for example, that the kinetics of the hand movements during a race can be decomposed into

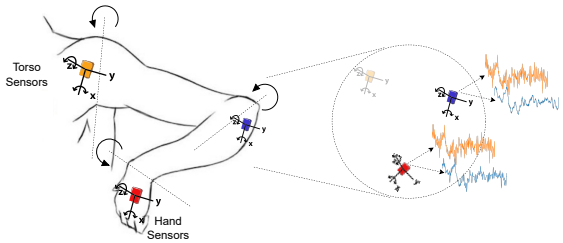

Fig. 1: Example of phenomena surrounded by a structured sensing environment. The hand sensor undergoes two types of movements. One is of the same nature as the torso and linked to the translational movement of the body. The other is linked to the movement of the hand locally relative to the body.

a circular movement (CM) of the hand relative to the shoulder and a translation movement (TM) associated with the whole body [23].

These characteristics pose significant challenges for decentralized machine learning as the data distribution across these devices and locations is skewed. Federated learning [22,16] is an appropriate framework that handles decentralized and distributed settings. In particular, the locally learned weights are aggregated into a central model during the conciliation phase. Decentralized machine learning suffers from objective inconsistency caused by the heterogeneity in local updates and by the interpretation of the locally collected data. Additional phenomena like the evolution of the local variables over time (concept drift) [15] or relativity of viewpoints (see Fig. 1) must also be considered.

Recent advances in machine learning literature, e.g., [31], seek the notions of invariance and symmetries within the phenomena of interest. Symmetry is one of the invariants that is leveraged for its powerful properties and its promising ability to drastically reduce the problem size [4,6,27] by requiring fewer training examples than standard approaches for achieving the same performance. Group theory provides a useful tool for reasoning about invariance and equivariance. For instance, in HAR [26,25], the acceleration recorded by the device held in hand must be corrected by the invariant related to the movement of the hand relative to the body so that the acceleration data related to the whole body is accurate. More generally, when the sensors are placed in a structured environment that exhibits regular dependencies between the locations of the sensors, it is possible to devise models of data transformations to reduce biases such as position biases. These models correspond to automatic changes in data representation to project them onto the same space while minimizing the impact of structure and location on the final data.

In this paper, we propose a novel approach that abstracts the exact context surrounding the data generators and hence improves decentralized machine learning. Local learners are trained to decompose the learned representations into (i) universal components shared across devices and locations and (ii) local components which capture the specific device- and location-dependent context.

We introduce the notion of relativity between data generators and model it via the special Euclidean group, denoted by $SE(3)$, which encompasses arbitrary combinations of translations and rotations. The relative contribution of a data generator in the description of the phenomena of interest is expressed using elements of this group and used to constrain the separation process. In particular, building on the symmetry-based disentanglement learning [12], the symmetry structure induced by the relative data generators is reflected in the latent space. This allows us to further leverage the notion of sharing which is reflected into the conciliation process of the decentralized learning setting by promising improvements. Comprehensive experimental evaluations are conducted to assess the effectiveness of the proposed approach. Obtained results demonstrate the superiority of the proposed method over more advanced solutions.

The main contributions of the paper are: (i) a novel approach that leverages additional knowledge in the terms of symmetries and invariants that emerge in these kinds of environments. These symmetries and invariants are explicitly represented in the form of group actions and incorporated into the learning process; (ii) a proposition of separation process of the data into universal and position-specific components improves collaboration across the decentralized devices materialized by the conciliation (or aggregation) process; (iii) extensive experiments on two large-scale real-world wearable benchmark datasets featuring structured sensing environments. Obtained results are promising noticeably in terms of the quality of the conciliation which open-up perspectives for the development of more efficient collaboration schemes in structured environments.

2 Background and Motivation

Here we provide a background on decentralized machine learning approaches and highlight their key principles. Then we review the impact of the various contexts surrounding the distributed data generators on the learning process in real-world IoT applications and *a priori* knowledge can be leveraged to deal with this challenge.

2.1 IoT Deployments

We consider settings where a collection $\mathcal{S}$ of M sensors (also called data sources), denoted $\{s_1, \ldots, s_M\}$, are positioned respectively at positions $\{p_1, \ldots, p_M\}$ on the object of interest, e.g., human body. Each sensor s_i generates a stream $\mathbf{x}^i = (x_1^i, x_2^i, \ldots)$ of observations of a certain modality like *acceleration*, *gravity*, or *video*, distributed according to an unknown generative process. Furthermore, each observation can be composed of channels, e.g. three axes of an accelerometer. The goal is to continuously recognize a set of human activity target concepts $\mathcal{Y}$ like *running* or *biking*. In the case of the SHL dataset, the data are generated from 4 smartphones, carried simultaneously at (*hand*, *torso*, *hips*, and *bag* body locations. Sensors distributed in various positions of the space provide rich perspectives and contribute in different ways to the learning process, and the

decentralization of the sensors has the potential to offer better guarantees of the quality of the generalization.

2.2 Decentralized Machine Learning

In the decentralized machine learning setting, a set of M clients, each corresponding to a sensor of the above IoT deployment, aim to collectively solve the following optimization problem:

$$\min_{w \in \mathbb{R}^d} \left\{ F(w) := \sum_{p=1}^{M} \alpha_p \cdot f_p(w_p) \right\}, \tag{1}$$

where $f_p(w) = \frac{1}{n_p} \sum_{\zeta \in D_p} \ell_p(x; \zeta)$ is the local objective function at the p-th client, with ℓ_p the loss function and ζ a random data sample of size n_p drawn from local dataset D_p according to the distribution of position p. At each communication round r, each client runs independently τ_p iterations of the local solver, e.g., stochastic gradient descent, starting from the current global model (set of weights) $w_p^{(r,0)}$ until the step $w_p^{(r,\tau_p)}$ to optimize its own local objective. Then the updates of a subset of clients are sent to the central server where they are aggregated into a global model. Only parameter vectors are exchanged between the clients and the server during communication rounds while raw data are kept locally which complies with privacy-preserving constraints. Various algorithms were proposed for aggregating the locally learned parameter vectors into a global model, including [22] which updates the shared global model as follows:

$$w^{(r+1,0)} - w^{(r,0)} = -\sum_{p=1}^{M} \alpha_p \cdot \eta \sum_{k=0}^{\tau_p - 1} g_p(w_p^{(r,k)}) \tag{2}$$

where $w_p^{(r,k)}$ denotes the model of client p after the k-th local update in the r-th communication round. Also, η is the client learning rate and g_p represents the stochastic gradient computed over a mini-batch of samples.

2.3 IoT Deployments and Impact of the Context

Long lines of research studied the impact of the varying contexts on machine learning algorithms and showed their fragility to viewpoint variations [14]. For example, basic convolutional networks are found to fail when presented with out-of-distribution category-viewpoint combinations, i.e., combinations not seen during training. Similarly, in activity recognition, the diversity of users, their specific ways of performing activities, and the varying characteristics of the sensing devices have a substantial impact on performances [29,10]. In these cases, the conditional distributions may vary across clients even if the label distribution is shared [15]. In decentralized approaches, several theoretical analyses bound this drift by assuming bounded gradients [36], viewing it as additional noise [17],

or assuming that the client optima are ϵ-close [19]. As a practical example, SCAFFOLD [16] tries to correct for this client-drift by estimating the update direction for the server model ($\mathbf{c}$) and the update direction for each client $\mathbf{c}_p$. Then, the difference $(\mathbf{c} - \mathbf{c}_p)$ is used as the estimator of the client-drift which is used to correct the local update steps. The local models are, then, updated as $w_p^{(r+1,0)} - w_p^{(r,0)} = -\eta \cdot (g_p(w_p) + \mathbf{c} - \mathbf{c}_p)$.

The impact of varying contexts is not limited to a skewed distribution of labels but is rather predominantly related to the aspects of the phenomenon being captured by the sensing devices depending on their intrinsic characteristics and locations. Depending on their disposition w.r.t. to the phenomena of interest, the sensing devices generate different views of the same problem. The heterogeneity brought by these configurations in terms of views is beneficial but must be explicitly handled. Reconciling the various perspectives offered by these deployments using decentralized learning approaches requires several relaxations limiting their potential capabilities when the impact of the context on the data generation process is essential. Indeed, how to reconcile these different points of view which can potentially be redundant or even seemingly contradictory to each other? When additional knowledge is available about the structure of the sensing environment, these challenges can be handled efficiently.

2.4 Relativity of Viewpoints in Structured Sensing Environments

Very often, knowledge about the relative geometry of the sensing devices and domain models describing the dynamics of the phenomena is available and can be leveraged and incorporated into the learning process. For example, the spatial structure of the sensors deployment and the induced views, sensors capabilities and the perspectives (views) through which the data is collected (sensing model, range, coverage, position in space, position on the body, and type of captured modality) [1,33,11]. A long line of research work around activity recognition reviewed in, e.g. [34,9], has focused on the problem of optimal placement and combination of sensors on the body in order to improve *a priori* models' performance. Additionally, domain models derived from biomechanical studies like [23,3] are often used to describe body movements and the relative interactions between various body parts in a structured manner. Alternatively, considering the structure of the sensing devices explicitly during the learning process is more promising but challenging. An approach close to ours for the relativity of perspectives is that of [5] which describes the different perspectives by discrete subgroup of the rotation group.

Integrating these additional models into the learning process has promising implications noticeably on the conciliation process of decentralized machine learning algorithms: one can exhibit the relative contribution of the individual views to the bigger picture. The primary goal of this paper is to develop a robust approach that integrates knowledge about the structure of sensing devices in a principled way to achieve better collaboration.

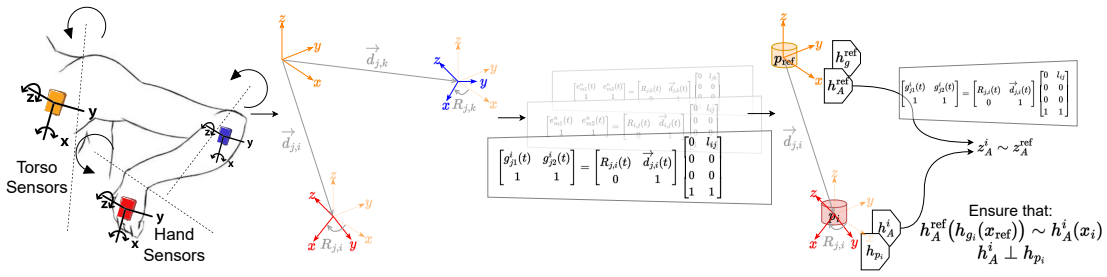

Fig. 2: Framework of the proposed approach. Explicitly representing the relative geometry of the decentralized devices and their symmetries using elements of the special Euclidean group $SE(3)$ and leveraging them to constrain the learning process with the goal of reducing the problem size and improving data efficiency.

3 FedAbstract Algorithm

We propose an original approach based on local abstraction of the position-specific artifacts and aggregation of universal components in the data. We leverage knowledge about the structure of the sensing deployment by representing the relative geometry of the sensing devices with group transformations. At a given decentralized location, there are three different elements that are learned: (1) the universal (or group-invariant) and (2) position-specific representations (§3.1), and (3) the group of relative geometry representation (§3.2). The generalization capabilities of the universal representation are improved collaboratively across the decentralized sensing devices via the conciliation (or aggregation) process (§3.3). Fig. 2 summarizes the proposed approach.

3.1 Learning Group-Invariant and Position-Specific Representations

The idea is to express the data generated from a decentralized device (e.g., hand sensors in the case of on-body sensor deployments) relative to the coordinate system of a referential (e.g., torso.) This way, the exact relative contribution of the sensing device is captured without the contextual artifacts. To do this, we have to capture the variations due to the relative location of the decentralized device w.r.t. a global coordinate system and capture invariant aspects that are shared across the devices. The latter aspects are universal components that are shared with the central model while the former ones are considered as specific components which add noise to the learning process, thus requiring to be discarded from it.

Invariance. A mapping $h(\cdot)$ is invariant to a set of transformations G if when we apply any transformation induced by g to the input of h, the output remains unchanged. A common example of invariance in deep learning is the translation invariance of convolutional layers. In the structured sensing environments considered here, the elements g of $SE(3)$ act on the spatial disposition of the

data generators and ultimately the data they generate: if we translate the data representation learned at sensor position p_i to position p_j, the representation remains unchanged. Formally, if $h : A \to A$, and G is a set of transformations acting on A, h is said to be invariant to G if $\forall a \in A$, $\forall g \in G$, $h(ga) = h(a)$.

We construct at the level of each client i a representation that maps the observation space X to a latent space V with $h_A : X \to V$ (universal) and $h_{p_i} : X \to V$ (position-specific). The universal representation has to remain invariant to the relative location of the decentralized nodes. We also ensure during the learning process that the universal and location-specific transformations are orthogonal to each other ($h_A \perp h_{p_i}$). In other words, we want these two transformations to capture completely different factors of variations in the data. To do that, we enforce h_{p_i} to be insensitive to the factors of variations linked to the representation h_A using representation disentanglement techniques. We use in our approach, a family of models based on variational autoencoders (VAEs) [18] for their ability to deal with entangled representations.

Learning h_A and h_{p_i} locally The data x_i captured at a given location i are generated from two underlying factors: one reflecting the position-specific components and the other the position-invariant (or universal) components. The task here is to learn these factors of variation, commonly referred to as learning a disentangled representation. In other words, we want these two transformations to capture completely different factors of variations in the data. To do that, we enforce h_{p_i} to be insensitive to the factors of variations linked to the representation h_A using representation disentanglement techniques. It corresponds to finding a representation where each of its dimensions is sensitive to the variations of exactly one precise underlying factor and not the others. Note that the inputs to h_A in the local learners are the raw sensory data x_i generated locally.

At this point, we are left with two alternatives for jointly learning the universal transformation h_A and the position-specific transformation h_{p_i} at the local learner level: (1) using a separate VAE for each transformation and training each one of them jointly using the raw sensory data as inputs; (2) using a single VAE and train it to automatically factorize the learned representation so that each axis captures specific components. Recent advances in unsupervised disentangling based on VAEs demonstrated noticeable successes in many fields using the β-VAE, which leads to improved disentanglement [13]. It uses a unique representation vector and assigns an additional parameter ($\beta > 1$) to the VAE objective, precisely, on the Kullback-Leibler (KL) divergence between the variational posterior and the prior, which is intended to put implicit independence pressure on the learned posterior. The improved objective becomes:

$$\begin{aligned}\mathcal{L}(x;\theta,\varphi) =& \mathbb{E}_{q_\varphi(z|x)}[\log p_\theta(x|z)] \text{ (autoencoder reconstruction term)} \\ &- \beta D_{KL}(q_\varphi(z|x)||p(z)) - \alpha D_{KL}(q_\varphi(z)||p(z)),\end{aligned}$$

where the term controlled by α allows to specify a much richer class of properties and more complex constraints on the dimensions of the learned representation

other than independence. Indeed, the proposed conciliation step is challenging due to the dissimilarity of the data distributions across the local learners, leading to discrepancies between their respective learned representations.

One way to deal with this issue is by imposing sparsity on the latent representation in a way that only a few dimensions get activated depending on the learner and activities. We ensure the emergence of such sparse representations using the appropriate structure in the prior $p(z)$ such that the targeted underlying factors are captured by precise and homogeneous dimensions of the latent representation. We set the sparse prior as $p(z) = \prod_d (1-\gamma)\mathcal{N}(z_d; 0, 1) + \gamma\mathcal{N}(z_d; 0, \sigma_0^2)$ with $\mathcal{N}$ is the Gaussian distribution. This distribution can be interpreted as a mixture of samples being either activated or not, whose proportion is controlled by the weight parameter γ [21].

Now, we have to represent the notion of data generators relativity and its induced symmetries in the form of group elements whose action on the data leaves the universal component of the learned representation invariant.

3.2 Relative Geometry for Data Generators

We model the relative geometry of sensors and the perspectives they provide via the special Euclidean group $SE(3)$. Let $\mathbf{x}^i$ and $\mathbf{x}^j$ be the stream of observations generated by the data sources s_i and s_j. At each time step t, the observations x_i and x_j generated by these data sources are related together via an element $g_j^i \in SE(3)$ of the group of symmetries, i.e., the observation x_i is obtained by applying g_j^i on x_j. Here, we want to learn a mapping h_{g_i} for each decentralized device, so that the biases that stem from the context (exact position) are corrected before its contribution is communicated to the global model.

Special Euclidean group $SE(3)$. The special Euclidean group, denoted by $SE(3)$, encompasses arbitrary combinations of translations and rotations. The elements of this group are called rigid motions or Euclidean motions and correspond to the set of all 4 by 4 matrices of the form $P(R, \overrightarrow{d}) = \left(\begin{smallmatrix} R & \overrightarrow{d} \\ 0 & 1 \end{smallmatrix}\right)$, with $\overrightarrow{d} \in \mathbb{R}^3$ a translation vector, and $R \in \mathbb{R}^{3\times 3}$ a rotation matrix. Members of $SE(3)$ act on points $z \in \mathbb{R}^3$ by rotating and translating them: $\left(\begin{smallmatrix} R & \overrightarrow{d} \\ 0 & 1 \end{smallmatrix}\right)\left(\begin{smallmatrix} z \\ 1 \end{smallmatrix}\right) = \left(\begin{smallmatrix} Rz + \overrightarrow{d} \\ 1 \end{smallmatrix}\right)$.

Relative geometry representation. Given a pair of sensing devices s_i and s_j located at positions p_i and p_j, each having its own local coordinate system attached to it. We represent the relative geometry of this pair by expressing each of the devices in the local coordinate system of the other (see Fig. 3). Similarly to [32], the local coordinate system attached to p_i is the result of a translation $\overrightarrow{d}_{j,i}$ and a rotation $R_{j,i}$, where the subscript j, i denotes the sense of the transformation being from p_j to p_i. While the translation corresponds to the alignment of the origins of the two coordinate systems, the rotation is obtained by rotating the global coordinate system such that the x-axis of the two coordinate systems coincide: $\begin{pmatrix} g^i_{j1}(t) & g^i_{j2}(t) \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} R_{j,i}(t) & \overrightarrow{d}_{j,i}(t) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & l_{ij} \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{pmatrix}$.

The relative geometry of the data generators is considered to be elements of $SE(3)$ and supposed to capture the transformations acting on the data generators. Without explicit information about the exact locations of the data generators, these transformations have to be learned. For this, we parameterize the transformation matrices used to represent the relative geometry of the data generators, with learnable weights. In particular, we parameterize as in [27] the n-dimensional representation of a rotation R as the product of $\frac{n(n-1)}{2}$ rotations, denoted $R^{v,w}$, each of which corresponds to the rotation in the v, w plane embedded in the n-dimensional representation. For example, a 3-dimensional representation has three learnable parameters, $g = g(\theta^{1,2}, \theta^{1,3}, \theta^{2,3})$, each parameterizing a single rotation, such as $R^{1,3}(\theta^{1,3}) = \begin{pmatrix} \cos\theta^{1,3} & 0 & \sin\theta^{1,3} \\ 0 & 1 & 0 \\ -\sin\theta^{1,3} & 0 & \cos\theta^{1,3} \end{pmatrix}$.

Learning h_A and h_g in the central server The referential learner (or central server) happens also to be a learner similar to the local learners. The main difference is that the referential learner is located in a particular position of the sensors deployment, i.e., the referential coordinate system, which imposes it to perform additional processing. Let's denote the referential learner with subscript ref (the orange data source in Fig. 3). The referential learner maintains the specific h_g's corresponding to each individual position of the sensors deployment and ensures that:

$$h_A(h_{g_i}(x_{\mathsf{ref}})) = h_A(x_i), \forall i \tag{3}$$

where h_{g_i} is the learned representation corresponding to the group action acting on the data x_i generated by the sensor located at position i and x_{ref} is the data generated by the sensor located at the referential point. The h_{g_i} transformation is learned by the referential learner using the raw data generated at the central server level. The constraint imposing the invariance, i.e., $h_A(h_{g_i}(x_{\mathsf{ref}})) = h_A(x_i), \forall i$, is the pivotal element that makes it possible to effectively learn this transformation.

By drawing a parallel with the construction of manifolds in latent spaces, this transformation can be interpreted as an operator projecting the data, generated by the data source positioned on ref, towards a latent space shifted by the action of the group elements so that the universal components learned by the transformation h_A (at the referential) coincide with those transformations $(h_A^i, \forall i)$ learned by the local learners attached to the other positions. h_g must therefore act on different subgroups of the latent space. We ensure that the learned universal transformation h_A is invariant to the action of the group $SE(3)$, i.e., $h_A(gx) = h_A(x), g \in SE(3)$. For this we map the group $SE(3)$ to a linear representation GL on V, i.e., $\rho : SE(3) \to GL(V)$. Our goal is to map observations to a vector space V and interactions to elements of $GL(V)$ to obtain a disentangled representation of the relative geometry.

As there are many different group representations (one for each position of the deployment of the sensors) at the referential learner's level, we have to ensure that the learned representation h_g acts on specific subspaces of the latent

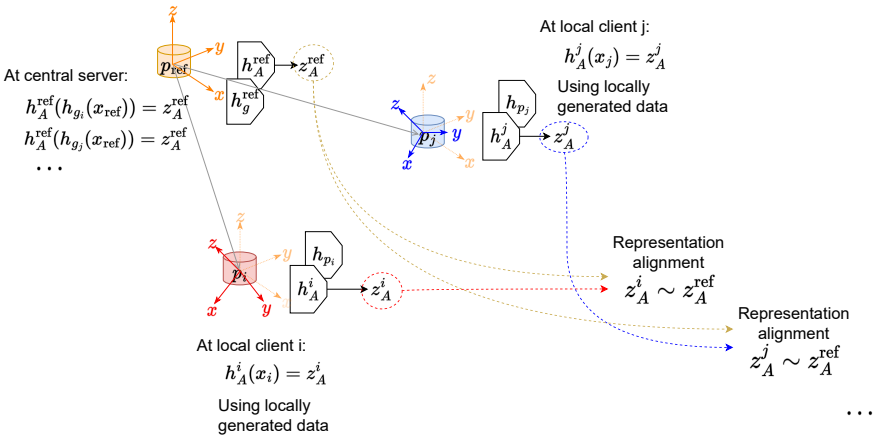

Fig. 3: Network architecture of FedAbstract. The local learners (red and blue) perform a set of updates on their proper version of the universal representation. The referential learner at position p_{ref} (in orange) maintains the specific h_g's corresponding to each individual position of the sensors deployment and ensures that: $h_A(h_{g_i}(x_{\text{ref}})) = h_A(x_i), \forall i$ where h_{g_i} is the learned representation corresponding to the group elements acting on the data x_i generated at position i and x_{ref} the data generated at the referential point. Notice that only gradient updates are shared to the central server and the data generated at a given location are processed exclusively by the local learner.

space. At the central server, each client is considered to generate a subgroup of relative geometry. During the learning process, each subgroup of the symmetry group is made to act on a specific subspace of the latent space. Formally, let $\cdot : G \times X \to X$ be a group action such that the group G decomposes as a direct product $G = G_1 \times G_2$. According to [12], the action is disentangled (w.r.t. the decomposition of G) if there is a decomposition $X = X_1 \times X_2$, and actions $\cdot_i : G_i \times X_i \to X_i, i \in \{1, 2\}$ such that: $(g_1, g_2) \cdot (v_1, v_2) = (g_1 \cdot_1 v_1, g_2 \cdot_2 v_2)$, where $\cdot$ denotes the action of the full group, and the actions of each subgroup as $\cdot_i$. An G_1 element is said to act on X_1 but leaves X_2 fixed, and vice versa. We end up here in the same situation as in the disentanglement of universal and position-specific components, i.e., either we use a separate VAE for each group transformation or a single one for all the groups with the additional constraint stating that the action of each subgroup act on specific regions of the latent space manifold and leave the other regions fixed. This can be achieved via clustering of the latent space using a Gaussian mixture prior [21] $p(z) = \sum_{c=1}^{C} \pi^c \prod_d \mathcal{N}(z_d | \mu_d^c, \sigma_d^d)$, with C the number of desired clusters and π^c the prior probability of the c-th Gaussian.

3.3 Conciliation Process

At the local learner's level, the proposed model is trained in an end-to-end fashion. The generalization capabilities of the representation h_A are improved via the conciliation process performed across the nodes of the deployment.

Algorithm 1: Multi-level abstraction of sensor position

Input : $\{\mathbf{x}^p\}_{p=1}^{M}$ streams of annotated observations

$w \leftarrow$ initWeights() ; % *Initialize global learner's weights*
distributeWeights(w, $\mathcal{S}$) ; % *Weights distribution*
while *not converged* **do**
  foreach *position* p **do**
    for $t \in \tau_p$ *steps* **do**
      Sample mini-batch $\{x_i^p\}_{i=1}^{n_p}$
      Evaluate $\nabla_{w_p}\ell(w_p)$ w.r.t. the mini-batch
        └ Subject to $J(z_A^p, z_A^{\mathsf{ref}})$ (e.g., correlation-based alignment [2])
      $w_p^{(t)} \leftarrow w_p^{(t-1)} - \eta\nabla_{w_p}\ell(w_p)$
      Ensure $h_A \perp h_{p_i}$ (see §3.1)
    end
    Communicate w_A (with $w_p = [w_A, w_{p_i}]$)
  end
  $w_A \leftarrow w_A + \sum_{p=1}^{M} \alpha_p \cdot \Delta w_A^p$; % *Central updates*
  Enforce group action disentanglement
end

Result: Globally shared universal representation h_A

Each local learner pursues its own version of the universal representation but has not to diverge from the referential universal representation h_A^{ref}, which constitutes a consensus among all local learners. After a predefined number of local update steps, we conduct a conciliation step (see the dotted arrows in Fig. 3). Each conciliation step t produces a new version of the referential learner $w_{\mathsf{ref}}^{(t)}$ and, a new version of the referential universal representation z_A^{ref}. The conciliation step has to be performed on the learned representations z_A^p via regularization, for example. In our approach, the conciliation step is performed via representation alignment, e.g., correlation-based alignment [2]. More formally, we instrument the objective function of the local learners with an additional term derived from the representation alignment [30]. The optimization problem (1) becomes:

$$\min_{w \in \mathbb{R}^d} \left\{ F(w) = \frac{1}{M} \sum_{p=1}^{M} \alpha_p (f_p(w_p) + \lambda J(z_A^p, z_A^{\mathsf{ref}})) \right\}, \tag{4}$$

where J is a regularization term responsible for aligning the locally learned universal components with the ones learned by the referential learner and $\lambda \in [0, 1]$ is a regularization parameter that balances between the local objective and the regularization term. Algorithm 1 summarizes the process of the proposed approach and Fig. 3 illustrates its bigger picture.

4 Experiments and Results

We perform an empirical evaluation of the proposed approach, consisting of two major stages: (1) we verify the effectiveness of the proposed approach in the HAR task via a comparative analysis which includes representative related baselines (§4.1); (2) we also conduct extensive experiments and ablation analysis to demonstrate the effectiveness of the various components of our proposed approach (§4.2).

Experimental setup. We evaluate our proposed approach on two large-scale real-world wearable benchmark datasets featuring structured sensing environments: SHL [8] and Fusion [28] datasets. We compare our approach with the following closely related baselines.

- **DeepConvLSTM** [24]: a model encompassing 4 convolutional layers responsible for extracting features from the sensory inputs and 2 long short-term memory (LSTM) cells used to capture their temporal dependence.
- **DeepSense** [35]: a variant of the DeepConvLSTM model combining convolutional and Gated Recurrent Units (GRU) in place of the LSTM cells.
- **AttnSense** [20]: features an additional attention mechanism on top of the DeepSense model forcing it to capture the most prominent sensory inputs both in the space and time domains to make the final predictions.
- **GILE** [26]: proposes to explicitly disentangle domain (or position)-specific and domain-agnostic features using two encoders. To constrain the disentanglement process, their proposed additional classifier is trained in a supervised manner with labels corresponding to the actual domain to which the learning examples belong. Here, we use the exact location of the data sources as domain labels.

To make these baselines comparable with our models, we make sure to get the same complexity, i.e., a comparable number of parameters. We use the f1-score in order to assess performances of the architectures. We compute this metric following the method recommended in [7] to alleviate bias that could stem from unbalanced class distribution. In addition, to alleviate the performance overestimation problem due to neighborhood bias, we rely in our experiments on meta-segmented partitioning.

4.1 Performance Comparison

We conduct extensive experiments to evaluate the performance of the proposed algorithm in the following two settings: activity recognition (or classification) task and representation disentanglement. For the activity recognition setting, Table 1 summarizes the performance comparison of the baselines in terms of the f1-score obtained on the SHL and Fusion datasets. Here we assess the usefulness of the separated components per se by leveraging them in a traditional discriminative setting. In other words, we take the learned representation and add a

Table 1: Recognition performances (f1-score) of the baseline models on different representative related datasets. Evaluation based on the meta-segmented cross-validation.

Model	Fusion	SHL (Acc.)	SHL
DeepConvLSTM	68.5 ±.002	64.4 ±.0078	65.3 ±.0206
DeepSense	69.1 ±.0017	64.8 ±.0033	66.5 ±.006
AttnSense	70.3 ±.0027	69.6 ±.0072	68.4 ±.03
GILE	71.7 ±.014	71.1 ±.035	69.0 ±.001
FedAbstract	75.7 ±.047	75.7 ±.047	77.3 ±.017

simple dense layer on top of it. This additional layer is trained to minimize classification loss while the rest of the circuit is kept frozen. Experimental results show that the proposed approach exhibits superior performance compared to the baselines. The proposed method achieves promising improvements in terms of f1-score over the baseline methods. In particular, our proposed approach improves recognition performances by approximately 7-9% on Fusion and SHL, while the improvement of attention-based methods is only about 1-2%. Compared to GILE, our approach shows consistent improvement on the considered configurations. This demonstrates that leveraging knowledge about the structure of the deployment, instead of simply using domain labels corresponding to the exact location of the data sources, improves disentanglement and ultimately activity recognition.

In the representation disentanglement setting, we assess the separation quality between the universal and position-specific components as well as those related to the actions of each subgroup. For this, the average latent magnitude computed for each dimension of the learned representations constitutes an appropriate measure. Fig. 4 illustrates the average latent magnitude computed for the group of relative geometry representation. It shows the activated latent dimensions depending on the subgroup of transformations (among Bag, Hand, and Hips) acting on the data sources. We can see in particular that specific dimensions are activated depending on the subgroup of transformations that are used to stimulate the learned representation. These dimensions are also independent of each other. Furthermore, in complementary experiments, one can observe the evolution of the dimensions of the central learner's latent representation where some of them are getting more activated than others, which is a sign of the emergence of the desired universal components shared across the learners.

4.2 Ablation Study

To demonstrate the generalization and effectiveness of each component of our proposed approach, we further design and perform ablation experiments on the SHL and Fusion datasets. We compare FedAbstract to FedAvg [22] and advanced solutions which try to correct for client-drift including SCAFFOLD [16]. FedAvg

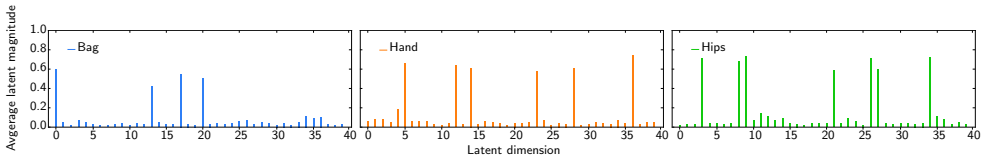

Fig. 4: Average latent encoding magnitude in the SHL dataset. It shows the repartition of the latent dimensions being activated between the different subgroups of transformations acting on the data sources (Bag, Hand, and Hips positions).

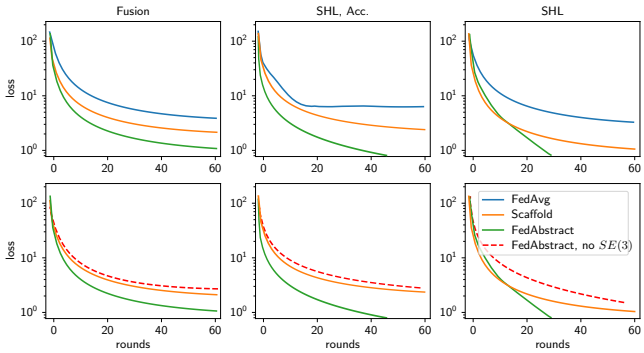

Fig. 5: Evolution of the loss during decentralized learning. (top) FedAbstract with both the relativity and decomposition constraints. (bottom) FedAbstract without the relativity representation constraints (FedAbstract, no $SE(3)$).

and SCAFFOLD do not perform explicit separation of the local data and thus constitute suitable baselines to assess the impact of each of FedAbstract's components. The experimental results illustrated in Fig. 5 (top) are obtained using FedAbstract with both the relativity and decomposition constraints. These results suggest that the evolution of the loss in the case of FedAvg gets slower as we increase the number of local steps, which corresponds to the common observation that client-drift increases proportionally to the number of local steps, hindering progress. At the same time, we observe that FedAbstract has excellent performance, slightly better than SCAFFOLD, suggesting a close connection between the estimate of the client-drift $\mathbf{c}_i$ and the position-specific components obtained via our proposed separation process.

Furthermore, we evaluate the effectiveness of explicitly representing the data generators' relativity via group actions while learning the universal and position-specific transformations. For this, we evaluate the performance of our proposed approach against a setting that does not specifically consider the relative geometry of the data generators. Basically, in this setting, the constraint imposing the relative geometry is not enforced during the learning process. Fig. 5 (bottom) illustrates the obtained results in terms of the loss evolution on both SHL and Fusion datasets. We notice that compared to the basic setting, enforcement of the relative geometry consistently improves the convergence by 5% on SHL

and 3% on Fusion. We see that these differences correspond to the gap between SCAFFOLD and our proposed approach. This demonstrates that the separation process constrained by the explicit representation of relativity ultimately leads to improving collaboration across the decentralized devices.

5 Conclusion and Future Work

In this work, we address the problem of decentralized learning in structured sensing environments. We propose a novel approach that leverages additional knowledge in terms of symmetries and invariants that emerge in these kinds of environments. These symmetries and invariants are explicitly represented in the form of group actions and incorporated into the learning process. Further, the proposed separation process of the data into universal and position-specific components improves collaboration across the decentralized devices materialized by the conciliation (or aggregation) process. Obtained results on activity recognition, an example of real-world structured sensing applications, are encouraging and open-up perspectives for studying more symmetries, invariants, and also equivariants that emerge in these environments. Future work also includes leveraging these symmetries and invariants from a theoretical perspective like Lie group and corresponding algebra, a special and large class of continuous groups that includes many valuable transformations like translations, rotations, and scalings and which also proposes a principled way for handling operations on the transformations such as composition, inversion, differentiation, and interpolation. The broader idea is that *universal* data is not directly accessible. On the other hand, it can be attained through various decentralized points of view. Collaboration is not a confrontation but rather the addition of relevant symmetries and complementary information from each viewpoint whose contribution can be determined precisely. The model we propose achieves this.

References

1. Aghajan, H., Cavallaro, A.: Multi-camera networks: principles and applications. Academic press (2009)
2. Andrew, G., Arora, R., Bilmes, J., Livescu, K.: Deep canonical correlation analysis. In: ICML. pp. 1247–1255. PMLR (2013)
3. Carollo, J., et al.: Relative phase measures of intersegmental coordination describe motor control impairments in children with cerebral palsy who exhibit stiff-knee gait. Clinical Biomechanics **59**, 40–46 (2018)
4. Caselles-Dupré, et al.: Symmetry-based disentangled representation learning requires interaction with environments. NeurIPS **32**, 4606–4615 (2019)
5. Esteves, C., Xu, Y., Allen-Blanchette, C., Daniilidis, K.: Equivariant multi-view networks. In: IEEE/CVF ICCV. pp. 1568–1577 (2019)
6. Finzi, M., et al.: Generalizing convolutional neural networks for equivariance to lie groups on arbitrary continuous data. In: ICML. pp. 3165–3176 (2020)
7. Forman, G., Scholz, M.: Apples-to-apples in cross-validation studies. ACM SIGKDD **12**(1), 49–57 (2010)

8. Gjoreski, H., et al.: The university of sussex-huawei locomotion and transportation dataset for multimodal analytics with mobile devices. IEEE (2018)
9. Hamidi, M., Osmani, A.: Data generation process modeling for activity recognition. In: ECML/PKDD. Springer (2020)
10. Hamidi, M., Osmani, A.: Human activity recognition: a dynamic inductive bias selection perspective. Sensors **21**(21), 7278 (2021)
11. Hamidi, M., Osmani, A., Alizadeh, P.: A multi-view architecture for the shl challenge. p. 317–322. UbiComp-ISWC (2020)
12. Higgins, I., Amos, D., Pfau, D., Racaniere, S., Matthey, L., et al.: Towards a definition of disentangled representations. arXiv:1812.02230 (2018)
13. Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., et al.: beta-vae: Learning basic visual concepts with a constrained variational framework. In: ICLR (2017)
14. Hsieh, K., Phanishayee, A., Mutlu, O., Gibbons, P.: The non-iid data quagmire of decentralized machine learning. In: ICML. pp. 4387–4398 (2020)
15. Kairouz, P., McMahan, H.B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A.N., et al.: Advances and open problems in federated learning. arXiv:1912.04977 (2019)
16. Karimireddy, S.P., et al.: Scaffold: Stochastic controlled averaging for federated learning. In: ICML. pp. 5132–5143 (2020)
17. Khaled, A., Mishchenko, K., Richtárik, P.: Tighter theory for local sgd on identical and heterogeneous data. In: AISTATS. pp. 4519–4529 (2020)
18. Kingma, D., Welling, M.: Auto-encoding variational bayes. arXiv:1312.6114 (2013)
19. Li, T., Sahu, A.K., Zaheer, M., Sanjabi, M., Talwalkar, A., Smith, V.: Federated optimization in heterogeneous networks. MLSys **2**, 429–450 (2020)
20. Ma, H., Li, W., Zhang, X., Gao, S., Lu, S.: Attnsense: Multi-level attention mechanism for multimodal human activity recognition. In: IJCAI. pp. 3109–3115 (2019)
21. Mathieu, E., Rainforth, T., Siddharth, N., Teh, Y.W.: Disentangling disentanglement in variational autoencoders. In: ICML. pp. 4402–4412 (2019)
22. McMahan, B., et al.: Communication-efficient learning of deep networks from decentralized data. In: AISTATS. pp. 1273–1282 (2017)
23. Melendez-Calderon, A., Shirota, C., Balasubramanian, S.: Estimating movement smoothness from inertial measurement units. bioRxiv (2020)
24. Ordóñez, F.J., Roggen, D.: Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. Sensors **16**(1), 115 (2016)
25. Osmani, A., Hamidi, M.: Reduction of the position bias via multi-level learning for activity recognition. In: PAKDD. pp. 289–302. Springer (2022)
26. Qian, H., et al.: Latent independent excitation for generalizable sensor-based cross-person activity recognition. In: AAAI. vol. 35, pp. 11921–11929 (2021)
27. Quessard, R., Barrett, T., Clements, W.: Learning disentangled representations and group structure of dynamical environments. NeurIPS **33** (2020)
28. Shoaib, M., Bosch, S., Incel, O.D., et al.: Fusion of smartphone motion sensors for physical activity recognition. Sensors **14**(6), 10146–10176 (2014)
29. Stisen, A., et al.: Smart devices are different: Assessing and mitigatingmobile sensing heterogeneities for activity recognition. In: ACM SenSys. pp. 127–140 (2015)
30. T Dinh, C., Tran, N., Nguyen, T.D.: Personalized federated learning with moreau envelopes. NeurIPS **33** (2020)
31. Vapnik, V., Izmailov, R.: Complete statistical theory of learning: learning using statistical invariants. In: COPA. pp. 4–40. PMLR (2020)
32. Vemulapalli, R., Arrate, F., Chellappa, R.: Human action recognition by representing 3d skeletons as points in a lie group. In: IEEE CVPR. pp. 588–595 (2014)
33. Wu, C., Khalili, A.H., Aghajan, H.: Multiview activity recognition in smart homes with spatio-temporal features. In: ACM/IEEE ICDSC. pp. 142–149 (2010)

34. Yang, J.Y., et al.: Using acceleration measurements for activity recognition. Pattern recognition letters **29**(16), 2213–2220 (2008)
35. Yao, S., et al.: Deepsense: A unified deep learning framework for time-series mobile sensing data processing. In: WWW. pp. 351–360 (2017)
36. Yu, H., et al.: Parallel restarted sgd with faster convergence and less communication. In: AAAI. pp. 5693–5700 (2019)