Transfer Learning for Human Motion Prediction: Improving Accuracy Under Data Scarcity

Marco Casarin, *Student Member, IEEE*, Michael Vanuzzo, *Student Member, IEEE*, Mattia Guidolin, *Member, IEEE*, Monica Reggiani, and Stefano Michieletto

***Abstract*—Human motion prediction (HMP) consists of anticipating future human poses based on motion history. While recent approaches achieved strong performance on benchmark datasets, their applicability to real-world scenarios remains limited due to the scarcity of context-specific motion data. In this work, we propose a unified, model-agnostic transfer learning (TL) framework for HMP to improve prediction accuracy under data-scarce target domains. The framework leverages pretraining on a large dataset of generic motions to learn a transferable motion prior, followed by systematic adaptation to context-specific actions using different TL techniques. We present a comprehensive evaluation protocol to characterize prediction accuracy across multiple representative HMP architectures, target datasets, and TL strategies and to evaluate training efficiency by introducing dedicated TL metrics. Beyond performance gains, we analyze the impact of source-target dataset distance and model complexity on TL effectiveness. The results show that TL consistently improves accuracy across all models and target datasets, reducing error up to 32.95% compared with training from scratch. In addition, TL enhances convergence speed, significantly reducing training epochs. These findings demonstrate TL as an effective and versatile approach for improving HMP under data scarcity, and provide methodological insights for its deployment in real-world context-specific applications.**

***Impact Statement*—This work presents the first comprehensive study of TL as a general approach for HMP to address data limitations in context-specific applications. By evaluating TL across multiple model architectures and motion domains, we demonstrate its effectiveness in improving prediction accuracy and training efficiency. These findings enable advances in the real-world applicability of HMP systems in domains such as collaborative robotics and healthcare, where limited data is a common challenge.**

Received 11 September 2025; revised 10 November 2025 and 13 January 2026; accepted 26 February 2026. Date of publication 4 March 2026; date of current version 31 July 2026. This work was supported in part by the European Union – Next Generation EU: DM118 (Italian PNRR – M4 C1, Invest 4.1); in part by the MICS (Made in Italy – Circular and Sustainable) Extended Partnership (Italian PNRR – M4 C2, Invest 1.3 – D.D. 1551.11-10-2022, PE00000004, CUP: C93C22005280001); and in part by the PRESENCE (anticiPatoRy bEhaviors for Safe and Effective humaN-robot CoopEration) project BIRD221598. This article was recommended for publication by Associate Editor Joo Hwee Lim upon evaluation of the reviewers' comments. *(Corresponding author: Marco Casarin.)*

The authors are with the Department of Management and Engineering, University of Padua, 36100 Vicenza, Italy (e-mail: marco.casarin.4@phd.unipd.it; michael.vanuzzo@phd.unipd.it; mattia.guidolin@unipd.it; monica.reggiani@unipd.it; stefano.michieletto@unipd.it).

This article has supplementary downloadable material available at https://doi.org/10.1109/TAI.2026.3670340, provided by the authors.

Digital Object Identifier 10.1109/TAI.2026.3670340

***Index Terms*—Context-specific applications, deep learning, human motion prediction (HMP), transfer learning (TL).**

I. Introduction

HUMAN motion prediction (HMP) consists of predicting future human movements given past motion history. This finds applications in diverse fields, including autonomous driving and crowd navigation [1], healthcare and rehabilitation [2], surveillance [3], and human–robot collaboration [4].

Early approaches for HMP relied on probabilistic methods [5]. These techniques were limited to predicting simple, periodic actions, such as walking or running. Deep learning (DL) models have now become the state-of-the-art due to their ability to learn more complex patterns from data [6]. However, effectively training these models requires a large volume of motion samples.

Recent years have seen the release of several human motion datasets, driven by the growing interest in understanding human behavior. These datasets primarily capture people performing generic actions in controlled environments (Fig. 1, in blue), typically empty spaces surrounded by cameras [6]. When HMP models are trained on these datasets, they perform effectively for generic movements but often struggle to generalize to context-specific scenarios [7]. This is because motion samples from the existing datasets rarely align with the actions and movements relevant to the context of interest. Real-world applications require accurate predictions to gain insights into the operational context. Unfortunately, context-specific datasets are often limited in size (Fig. 1, in yellow). One possible solution is to collect new, context-specific human motion data. However, this process is both time consuming and expensive [8]. Consequently, the use of HMP systems in real-world applications remains an open challenge [9].

While much of the existing HMP research has focused on developing novel models to improve accuracy on benchmark datasets, limited works have investigated strategies to enhance the applicability of HMP in scenarios where data is scarce. Addressing this gap, our work aims to improve the prediction accuracy when few motion samples are available. Instead of developing novel prediction models, the focus is on general approaches that can be applied independently of the specific architecture and context. In particular, the main objective of this research is to investigate the use of TL for tackling the lack of context-specific data [10]. It enables the improvement of model performance on a specific context (target domain)

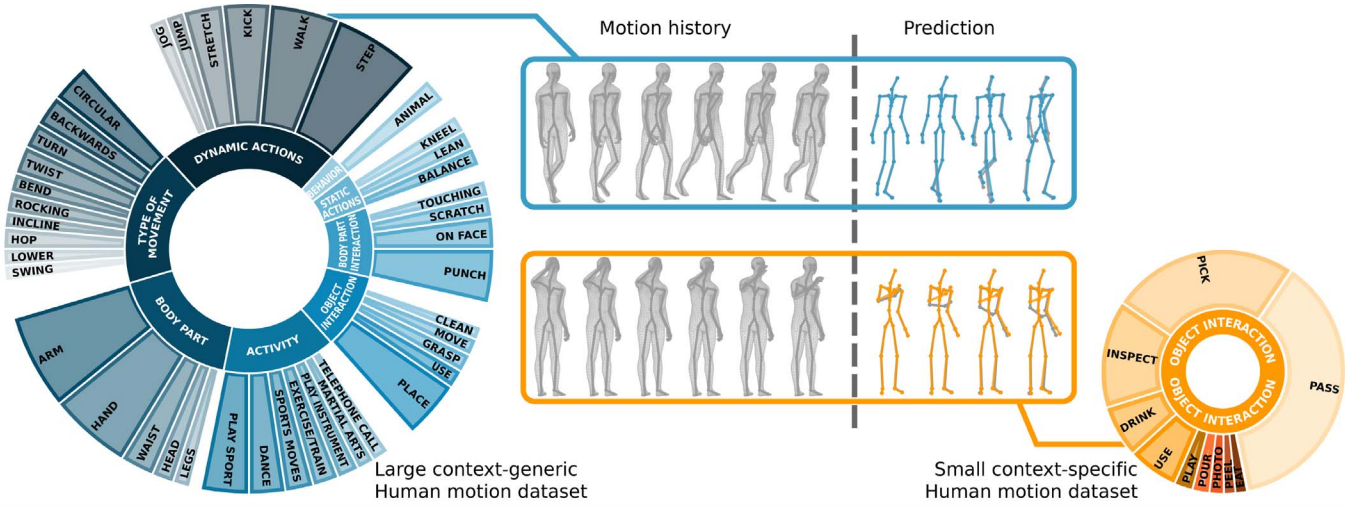

Fig. 1. Example of the current availability of human motion datasets for HMP. In blue, a typical large, context-generic dataset containing generic actions recorded in controlled environments. In yellow, the typical limited availability of context-specific motion for a particular HMP application. In the middle, a qualitative comparison of model accuracy when trained on these datasets. While the models tend to perform well on generic movements, they struggle with context-specific motions due to data scarcity.

by transferring knowledge from a different but related context (source domain).

In this work, we aim to answer the following question: "Is it beneficial to leverage prior knowledge about human motion from large, generic datasets when moving to context-specific motions with scarce training data?" Previous studies have explored Fine-Tuning strategies for HMP in limited experimental settings [11]. However, a general and systematic evaluation and analytical characterization of TL effectiveness across different architectures and target domains remains unexplored.

To characterize the generalizability of human knowledge transfer, we develop a dedicated TL framework and apply it to four prediction models that well represent the main HMP approaches. Additionally, we evaluate two TL methods on three target datasets. Each target dataset contains a limited number of motion samples from a specific set of human actions, reflecting real-world data-scarce scenarios. A large and diverse dataset, Archive of Motion Capture of Surface Shapes (AMASS) [12], is used for model pretraining. To analyze the effectiveness of TL, we analyze the prediction accuracy across the same model when applying TL and when training from scratch on the target dataset. We analyze the results based also on two additional key factors, dataset distance and model complexity. In addition to accuracy, we introduce dedicated metrics to analyze the advantages of TL in the training process.

The main contributions of this work are the following.

1) A unified, model-agnostic TL framework for HMP, enabling systematic comparisons between scratch training and TL methods across diverse architectures and target domains in data-scarce conditions.
2) Novel TL metrics for HMP, designed to quantify the impact of knowledge transfer and evaluate training efficiency.
3) A systematic analysis of TL effectiveness based on dataset distance and model characteristics, introducing Fréchet Inception Distance as an indicator for source-target transferability and characterizing the relationship between model complexity and TL efficiency.
4) Practical and methodological guidelines on both the application and validation of TL in HMP.

The rest of this article is organized as follows. Section II reviews the current approaches to HMP and related works on TL. Section III states the problem of HMP and describes our TL framework and its components, including the adopted HMP models, TL techniques, and evaluation metrics. Section IV presents the human motion datasets and the experimental pipeline. Section V reports and discusses the results. Finally, Section VI concludes the article.

II. Related Works

HMP has reached significant attention in recent years, as it plays a key role in enabling seamless interaction between humans and machines [6]. Early approaches for HMP relied on statistical methods, such as hidden Markov models [13], [14], Gaussian process models [15], [16], and restricted Boltzmann machines [17]. These methods are effective for modeling simple and repetitive movements, but struggle with complex motions [18]. In recent years, DL models have become the dominant approaches for HMP, demonstrating strong performance due to their ability to learn complex patterns from data [6].

A. Deep Learning Models for HMP

Recurrent neural networks (RNNs) have been widely adopted for HMP due to their ability to model temporal dependencies [5], [19], [20], [21]. Fragkiadaki et al. [19] introduced an encoder-recurrent-decoder (ERD) architecture combining nonlinear encoders and decoders with recurrent layers. Martinez et al. [5] proposed modeling first-order motion derivatives using gated recurrent unit (GRU) cells to reduce discontinuities in

the first prediction frame, improving smoothness in predictions. In [20], Pavullo et al. addressed the error accumulation along the kinematic chain by employing quaternion representations rather than joint angles to describe human poses. To mitigate convergence to static poses, which is typical in autoregressive RNNs, Wang et al. [21] incorporated pose velocities and temporal positional information. RNN-based models have proven effective in predicting human motion, although they do not explicitly account for the spatial structure of the body.

Graph convolutional networks (GCNs) have become a popular alternative, modeling the human skeleton as a graph, where nodes correspond to the skeleton joints and edges represent their connections [22], [23], [24], [25], [26]. In [22], Mao et al. proposed a GCN-based model leveraging motion attention to focus on relevant past movements. Other works introduced multiscale GCNs to extract and combine spatial features at multiple resolutions, allowing the model to focus on individual aspects of the human body at different levels [23], [24]. Li et al. [25] proposed a symbiotic multiscale GCN architecture combining structural graphs for physical constraints and actional graphs for action-based relations between joints. Instead of using predefined adjacency matrices, in [26] Yang et al. incorporated static, learnable, and adaptive adjacency matrices in their GCN architecture to capture both fixed and action-specific joint relations. While GCNs have achieved state-of-the-art results in HMP, they significantly increase model complexity and require heavy computational resources for their training.

More recently, multilayer perceptron (MLP)-based models have emerged as a lightweight yet effective alternative in HMP [27], [28]. These models avoid using recurrence and convolutional operations and rely solely on feedforward layers. Bouazizi et al. [27] proposed a model that separately encodes spatial and temporal features using MLP blocks, combined with a squeeze-and-excitation block [29]. In [28], Guo et al. proposed siMLPe, a very light model that exploits the discrete cosine transform for temporal encoding. Its simplicity and strong predictive capabilities have established siMLPe as a commonly adopted HMP baseline.

The integration of physics principles in HMP models has also been explored to improve physical plausibility in predicted motion sequences [30], [31], [32]. In the work of Maeda et al. [30], the authors introduced a motion synthesis technique to augment training data, imposing a physical policy to improve the realism of the generated sequences. Zhang et al. [31] incorporated physical signals such as contact forces and joint torques, computed via a physics simulator. More recently, in [32], Zhang et al. introduced an HMP architecture that combines data-driven predictions with a physics-based block for solving the Euler–Lagrange equations. Imposing physics principles helps in short-term accuracy, while longer predictions are guided by the data-driven block.

B. Context-Specific Adaptation in HMP

Previous works have addressed the adaptation of HMP models in the context of collaborative robotics [33], [34], [35]. Moon et al. [35] proposed a prediction model to enable fast adaptation to individual users in cooperative human–robot tasks. Their architecture includes adaptive parameters that can be tuned using a small number of motion samples specific to each person. In [33], the authors extended the siMLPe architecture by incorporating the human intention as an additional input to guide the prediction process. The intention is defined as cooperative or noncooperative, and it is classified by an auxiliary module. Liang et al. [34] proposed a context-aware model that integrates specific environmental information, such as the positions of working tools, to guide predictions for collaborative construction scenarios.

Instead of focusing on a specific context, other studies have proposed few-shot adaptation techniques to improve predictions on a generic set of target motions [7], [36]. Gui et al. [7] proposed a framework for rapid generalization to novel human actions using limited data. In particular, their approach combines a meta-learning strategy for a generic model initialization, followed by target adaptation using model regression networks. In [36], the authors introduced MoPredNet, an HMP framework that can dynamically select informative motions and adapt its internal representation from a small number of unseen sequences.

Despite their effectiveness, these approaches often consist of custom model architectures, making their integration with existing HMP models not possible without heavy modifications. In addition, they often rely on task-specific inputs, and their adaptation to other domains would require extensive changes and the integration of domain-specific features.

C. TL in Human Motion Modeling

TL is a model-agnostic approach that allows for improving accuracy on limited data by transferring knowledge from a source set of motion samples to a target set of movements [37]. TL techniques can be seamlessly integrated into existing HMP architectures, enabling adaptation to target motions without requiring the design of context-specific features.

Recent works leveraged TL in HMP-related tasks to address data limitations. In the domain of human pose estimation, where synthetic data are commonly adopted for training, the performance on real-world scenarios often deteriorate. To address this, Doersch et al. [38] proposed a TL strategy to bridge the gap between simulated and real data. Similarly, Zhang et al. [39] focused on lower-limb joint torque estimation from wearable sensors data. Their approach involved pretraining the neural network on multiple subjects and tasks, followed by fine-tuning on specific target domains.

Other works have explicitly adopted TL for predicting human motion trajectories under varying contexts [40], [41]. Ballan et al. [40] used TL to improve trajectory predictions in unseen contexts. In [41], Akabane et al. addressed the lack of data for their mobile robotics application by pretraining their model on a large collection of generic human motion trajectories.

D. Comparison With Prior Work

Casarin et al. [11] conducted a preliminary investigation into the use of TL in HMP. However, the proposed approach was specifically designed for fine-tuning a single RNN-based model, limiting the generalization of the study. Moreover, the

evaluation was based on a single accuracy metric, leaving the critical aspects about knowledge transfer, domain shift, and training efficiency unexplored.

In this work, we overcome these limitations by establishing a systematic, model-agnostic framework for TL in HMP. We adopt three HMP metrics to evaluate both positional and angular accuracy, and we introduce three novel TL metrics to quantify the impact of pretrained knowledge and training efficiency. Furthermore, we investigate the correlation of TL gains with model architecture, model complexity, and dataset distance. In our study, we perform an extensive and systematic experimental evaluation including four distinct HMP models, which are representative of the main paradigms in HMP, and multiple TL strategies. As a result, this work provides robust and generalizable insights into the effectiveness of TL in HMP, and establishes a methodological foundation for its deployment and validation in real-world data-scarce scenarios.

III. Methods

In this work, we develop a dedicated framework to apply different TL strategies to multiple models and target datasets, allowing to investigate TL under varying conditions. Our framework includes an extensive evaluation protocol to compare TL accuracy with the scratch baseline. Moreover, we introduce TL metrics to evaluate training performance.

This section formulates the HMP problem and introduces our TL framework. Then, the prediction models used in the experimental study are presented, as well as the TL approaches. Finally, we describe the accuracy metrics and provide details about the TL metrics definitions.

A. Problem Statement

In HMP, human movement is modeled as a discrete-time sequence of poses. Each pose at time t is represented as $x_t = (e_1, e_2, \ldots, e_J) \in \mathbb{R}^{J \times D}$, where J is the number of joints, and D is the dimensionality of each joint descriptor $e_j \in \mathbb{R}^D$ (e.g., rotation of the joint or $3D$ coordinates of its keypoint). The HMP problem can be stated as follows: given as input an observed sequence of N past human poses $X_{1:N} = [x_1, \ldots, x_N]$, the goal is to predict the future sequence of M poses $X_{N+1:N+M} = [x_{N+1}, \ldots, x_{N+M}]$. Predictions up to 400 ms and 1000 ms are commonly used to evaluate short-term and long-term performance, respectively [6].

B. TL Framework in HMP

The proposed framework (Fig. 2) is designed to compare the performance on the target dataset of multiple HMP models when trained from scratch and when applying TL. For the scratch method, we directly train the model on the small, context-specific target dataset, representing the standard approach in HMP. When applying TL, instead, the model is first pretrained on a large, context-generic source dataset to learn prior knowledge on human motion. Then, the model is tuned on the target dataset by exploiting pretrained knowledge to improve context-specific performance.

To assess TL effectiveness, we evaluate and compare the performance of the two approaches by considering two aspects: 1) prediction accuracy, measured using standard HMP metrics; and 2) training efficiency, quantified using custom TL metrics defined in this work. The proposed framework can be applied with different prediction models, TL techniques, or target contexts, enabling extensive analyses across diverse HMP settings.

In the subsequent sections, we provide details about each component of our framework, including the selected HMP models, TL approaches, and evaluation metrics.

C. HMP Models

To investigate the generalizability of TL across different architectures, we selected four HMP models that well represent the main paradigms currently used: 1) RNNs; 2) GCNs; 3) MLPs; and 4) physics-informed approaches. This allows for a comprehensive evaluation of the effectiveness of TL regardless of the underlying model class. In the following, we introduce the selected models and describe their main features.

1) Position-Velocity Recurrent Encoder–Decoder (PVRED): PVRED [21] is based on RNNs with GRU cells to mitigate the vanishing of the gradient. It extends the recurrent encoder–decoder (RED) architecture [5] by incorporating pose velocities and their positional embeddings, rather than relying exclusively on human pose information. The velocities help encode the amount of motion between consecutive poses, while positional embeddings provide the temporal context of each pose within the sequence. As a recurrent architecture, PVRED is highly effective in modeling temporal dependencies. Moreover, the integration of pose velocities helps to prevent predictions from converging to a static pose.

2) History Repeats Itself (HRI): HRI [22] belongs to the class of graph-based neural models and leverages GCN layers with learnable adjacency matrices to capture spatial dependencies among body parts. In addition, HRI incorporates an attention mechanism to identify common patterns in the motion sequences. In fact, HRI explicitly incorporates the hypothesis that human movements often repeat over time, sharing common motion primitives. The attention mechanism, combined with the learnable adjacency matrix, makes HRI a powerful HMP model capable of learning relevant motion patterns.

3) siMLPe: siMLPe [28] is a very light but effective prediction model based on MLP blocks. Its architecture consists solely of fully connected layers, along with layer normalization and transpose operations. siMLPe exploits some common techniques that have been proven effective for HMP. It uses discrete cosine transform to encode temporal structure, and it predicts the residual joints displacement rather than their absolute position. Additionally, it includes motion velocity in the loss optimization to improve long-term predictions. Despite the simplicity of its architecture, siMLPe achieved competitive performance, and it is now commonly adopted as a baseline in HMP.

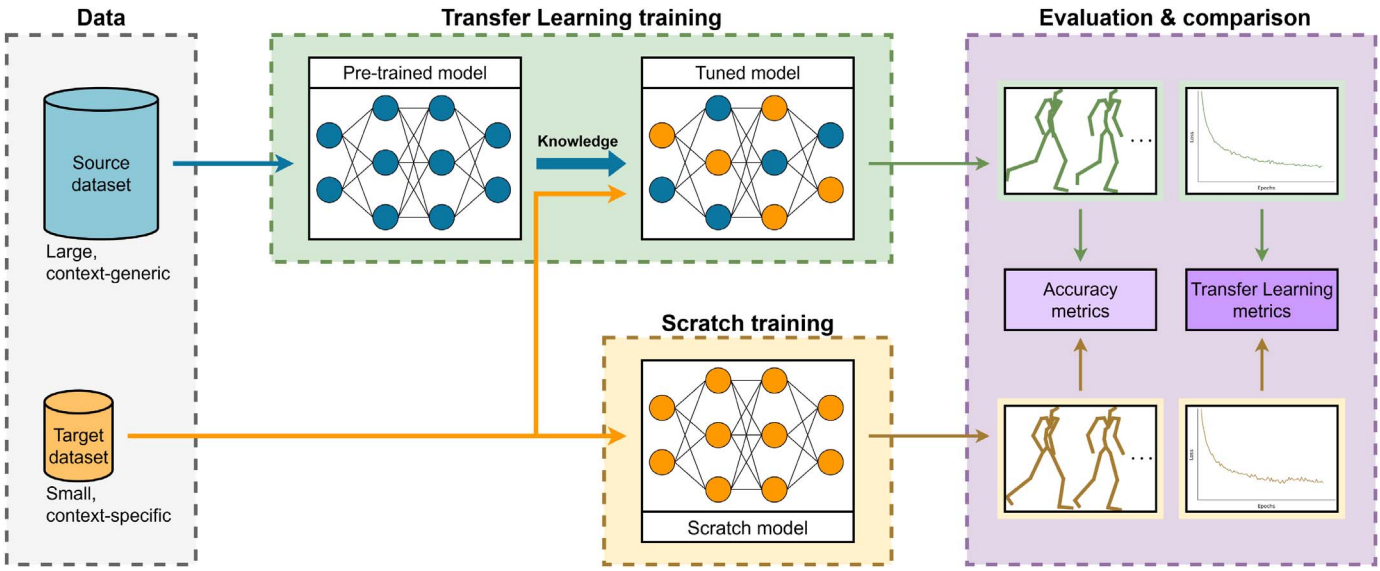

Fig. 2. TL evaluation framework in HMP. We compare the performance of different prediction models when trained from scratch and when applying TL. Evaluation includes prediction accuracy and TL metrics.

4) Physics for HMP (PhysMoP): PhysMoP [32] integrates physics modeling into DL by embedding a physics-based prediction block that captures human dynamics. This block solves the Euler–Lagrange equations, and it is effective for short-term predictions. A data-driven block based on MLP layers is also trained to guide long-term predictions. Finally, a fusion block balances the physics-based and data-driven contributions along the output sequence. PhysMoP provides a strong physical inductive bias, improving predictions plausibility and coherence. Therefore, it represents a complex, hybrid model that tests the TL ability to transfer both learned dynamics and physical priors.

By including models with fundamentally different assumptions—sequential modeling, spatial graphs, simple feedforward structures, and explicit physics—we ensure a broad evaluation of TL effectiveness across a representative spectrum of HMP approaches.

D. TL Approaches

To ensure broad applicability, this study focuses on TL approaches that do not depend on specific model structures or internal representations. Following the survey by Iman et al. [37], we selected two of the most widely adopted techniques: fine-tuning (FT) and freezing and adding new layers (FAL). Both methods follow a two-phase paradigm: 1) an initial pretraining phase, where the model is trained on a large, context-generic source dataset; and 2) a subsequent adaptation phase, where the learned motion priors are transferred to the target dataset.

1) Fine-Tuning (FT): In FT, the weights of the target model are initialized using those obtained during pretraining on the source dataset. This initialization transfers the knowledge of human motion learned during pretraining to the target model, providing a strong prior. After initialization, the model is then further trained on the target dataset to adapt its predictions to the specific motion characteristics. With FT, the neural network architecture and its parameters remain unchanged, with the exception of the learning rate, which we reduced by a factor of 10, as a common practice. This lower learning rate plays a critical role in mitigating the risk of catastrophic forgetting, a phenomenon where the model rapidly overwrites pretrained weights, leading to a loss of previously acquired knowledge [37]. By limiting the magnitude of parameter updates, FT encourages gradual adaptation, enabling the model to specialize on the target domain while preserving relevant information learned from the source.

2) Freezing and Adding New Layers (FAL): Similar to FT, in FAL the target model is first initialized with weights obtained through pretraining. However, instead of immediately updating all parameters, the pretrained layers are initially frozen, and a newly randomly initialized linear layer is added on top of the neural network. Only this layer is initially trained, forcing the model to preserve the pretrained knowledge and adapt to the target data using a minimal number of trainable parameters. This induces maximum retention of prior knowledge and reduces the risk of overfitting in small target dataset. Once the validation performance converges, all layers are unfrozen and the entire network is trained on the target actions. As in FT, the learning rated is reduce by a factor of 10, following common practice [37]. This stage allows for deeper adaptation of the entire model, refining both the newly added and previously frozen layers to better align with the target. Compared with FT, in FAL the pretrained knowledge is preserved more effectively and the adaptation is introduced in a more controlled manner.

E. Accuracy Metrics

Following established practices in HMP studies [5], [21], [22], [42], standard metrics are adopted to quantify prediction

accuracy over time. These metrics evaluate both spatial deviation and angular discrepancies between predicted and ground-truth poses. Let J represent the number of joints, while K the total number of test sequences.

1) Mean Per Joint Position Error (MPJPE): The MPJPE metric evaluates the 3D coordinates accuracy of the predicted skeleton keypoints by computing the average Euclidean distance between predicted and ground-truth positions

$$\mathrm{MPJPE_t} = \frac{1}{K \cdot J} \sum_{k,j} |\hat{p}_{t,k,j} - p_{t,k,j}|_2 \tag{1}$$

where $\hat{p}_{t,k,j}$ and $p_{t,k,j}$ denote the predicted and ground truth 3D coordinates of the keypoint j, respectively, at frame t in sequence k.

2) Mean Angular Error (MAE): The MAE metric calculates the Euclidean distance between the predicted and the ground truth joint angles. It is defined as follows:

$$\mathrm{MAE_t} = \frac{1}{K} \sum_{k} \|\hat{x}_{t,k} - x_{t,k}\|_2 \tag{2}$$

where $\hat{x}_{t,k}$ and $x_{t,k}$ are respectively the predicted and the ground truth poses, for frame t in sequence k.

3) Joint Angle Difference (JAD): JAD compares the minimum rotation required to align each predicted joint with the corresponding ground truth

$$\mathrm{JAD_t} = \frac{1}{K \cdot J} \sum_{k,j} \arccos\left(\frac{Tr(R^{\hat{x}_{t,k,j}}_{x_{t,k,j}}) - 1}{2}\right) \tag{3}$$

$Tr(\cdot)$ denotes the trace of a matrix, while $\hat{x}_{t,k,j}$ and $x_{t,k,j}$ represent respectively the predicted and ground truth angles for joint j at frame t in sequence k. The rotation matrix $R^{\hat{x}_{k,t,j}}_{x_{k,t,j}}$ expresses the minimum rotation required to align the predicted joint with the corresponding ground truth.

F. TL Metrics

We introduce dedicated metrics to assess the training efficiency with TL compared with training a model from scratch (SC). These metrics capture key aspects of the training process, including the convergence speed and the impact of pretrained knowledge on the training loss [43]. A visual representation of the TL metrics is provided in Fig. 3. In the following, let $e_{SC}(t)$ and $e_{TL}(t)$ denote the validation losses at epoch t for the scratch and TL models, respectively.

1) Jump-Start Performance (JSP): The JSP metric quantifies the impact of pretrained knowledge by comparing the validation loss after the first training epoch $e(1)$

$$\mathrm{JSP} = \frac{e_{SC}(1) - e_{TL}(1)}{e_{SC}(1)} \times 100. \tag{4}$$

The scratch model is randomly initialized, while the TL model begins with pretrained weights. Therefore, the loss difference at the start of training reflects the contribution of the pretrained knowledge.

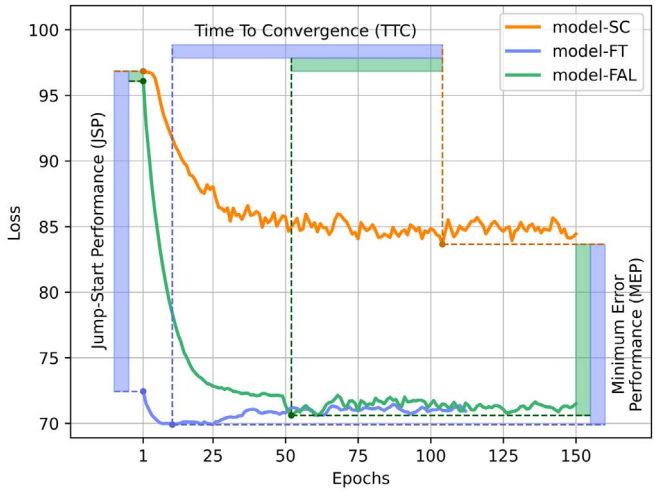

Fig. 3. Visual representation of the TL metrics. JSP highlights the contribution given by the pretrained knowledge by comparing the loss improvement after the first training epoch. MEP shows the best improvement in the overall validation loss when using TL. TTC measures the convergence speed showing how fast FT and FAL are compared with SC.

2) Minimum Error Performance (MEP): The MEP metric measures the minimum loss improvement when using TL

$$\mathrm{MEP} = \frac{\min(e_{SC}) - \min(e_{TL})}{\min(e_{SC})} \times 100 \tag{5}$$

where $\min(e_{SC})$ and $\min(e_{TL})$ indicate the lowest validation error achieved during training by the scratch and TL models, respectively.

3) Time to Convergence (TTC): The TTC metric highlights the relative convergence speed of TL compared with scratch

$$\mathrm{TTC} = \frac{t_{SC} - t_{TL}}{t_{SC}} \times 100 \tag{6}$$

where t_{SC} and t_{TL} represent the scratch and TL convergence epochs, respectively.

IV. Experimental Setup

This section introduces the experimental details of our work, including dataset descriptions, preprocessing steps, and training configurations.

A. Datasets

In our experiments, we used the AMASS [12] dataset, which unifies over 20 different motion capture databases within the Skinned Multiperson Linear Model (SMPL) [44] representation. We selected a subset of the AMASS datasets as the source data for model pretraining, while three additional datasets were used as distinct target domains. The source datasets were chosen based on the following criteria: 1) the motion content is broad and not context-specific; 2) recordings have a minimum framerate of 25 fps, avoiding the need for temporal upsampling; and 3) at least 90% of motion sequences are longer than 3 s, allowing the extraction of samples with 2 s of input and 1 s of prediction. The selected source datasets from AMASS are the following:

BMLmovi [45], BMLrub [46], EyesJapanDataset [47], HDM05 [48], MoSh [49], TotalCapture [50], CMU [51], EKUT [52], SFU [53], and Transitions [12].

In contrast to the generic source domain, the target datasets were chosen to represent context-specific actions, resembling real-world HMP applications and enabling the assessment of TL under domain shift. To this end, we selected GRAB [54] and DanceDB [55]. GRAB features subjects interacting with objects while stationary, with motion primarily limited to the upper body. This differs from the typical actions in human motion datasets, where leg movements are dominant, like in walking or jumping. DanceDB, instead, contains wide-ranging, nonperiodic full-body movements from dance performances, thus consisting of high kinematic variability. To provide a more comprehensive exploration of TL in HMP, we also included the ACCAD [56] dataset in the target domains. In contrast to GRAB and DanceDB, the actions in ACCAD are closer to the source motions, enabling the analysis of TL under reduced domain shift.

B. Preprocessing

We performed the following steps to preprocess both source and target data. First, following previous works [21], [22], [28], [32], all motion sequences were resampled to a common value of 25 fps. Then, we extracted fixed-length motion windows to handle the significant variability in sequence durations. Specifically, each sequence was segmented into 3 s (75 frames) overlapping windows with a stride of 50 frames, resulting in motion samples with 2 s (50 frames) of history and 1 s (25 frames) for the prediction. The chosen stride ensures that the 2 s input of consecutive windows does not overlap temporally, avoiding redundancy. To isolate the impact of motion context rather than data volume, the target datasets—originally of varying lengths—were truncated to a uniform size. This allows for a fair comparison across target domains and to emphasize the influence of action variability in TL effectiveness. Overall, the source dataset contains approximately 2137 min of motion sequences, while the target datasets 23 min, reflecting the limited data availability typical of HMP applications. Finally, the datasets were split into training (80%), validation (10%), and test (10%) sets.

C. Experimental Settings

Using our framework, we evaluated the effectiveness of TL in HMP by comparing models trained from scratch against those trained using fine-tuning (FT) or freezing and adding new layers (FAL). All four models were trained and tested using the same motion sequences sampled at 25 fps, with an input length $N = 50$ (2 s) and an output length $M = 25$ (1 s). When training from scratch, we retained the original model configurations as proposed in the respective works. The same settings were used for FT and FAL, except for the learning rate, which was reduced by a factor of 10 to mitigate catastrophic forgetting. For each experiment, TL metrics were computed from the validation loss, while accuracy metrics were calculated on the test set using the best model checkpoint.

V. Results and Discussion

This section presents and discusses the experimental results. First, we provide a quantitative evaluation of TL effectiveness using the HMP metrics, analyzing also the impact of datasets distance and model complexity. Next, we discuss the training performance based on the TL metrics. Finally, we show qualitative examples.

A. Accuracy Evaluation

For each accuracy metric, we report the prediction error at 120 ms, 200 ms, and 400 ms for short-term predictions, and at 600 ms and 1000 ms for long-term predictions. Additionally, we show the average improvement across the entire prediction horizon ($M = 25$) when using TL compared with the scratch baseline. We report accuracy results on additional TL approaches in the Supplementary Material.

1) MPJPE Evaluation: Table I shows the MPJPE results. Overall, TL approaches consistently improved prediction accuracy compared with training from scratch, particularly for short-term predictions. In DanceDB, PhysMoP-FAL reduced the error at 120 ms by 50% compared with PhysMoP-SC, achieving an even bigger gain in ACCAD. Long-term accuracy also improved significantly, demonstrating that TL helps to prevent diverging predictions compared with scratch models. Considering the average improvements, even the smallest gain of 10.13% for PVRED-FAL in DanceDB highlights a strong advantage of TL across the entire prediction horizon. The highest gain was achieved by PhysMoP when using FT in ACCAD, with a 32.95% average increase.

Both FT and FAL consistently outperformed the scratch baseline across different models and targets, thus confirming the generalization effectiveness of TL in HMP. Although FT proved slightly more effective than FAL, both TL methods achieved comparable results.

2) MAE Evaluation: In Table II, we present the MAE results for PVRED and PhysMoP. These two models are trained on an angle-based skeleton representation, while HRI and siMLPe use 3D coordinate notation. Therefore, applying inverse kinematics to obtain the corresponding rotations would result in nonunique solutions and, consequently, in a comparison that is not meaningful.

Both models showed strong improvements over the scratch baseline when using TL methods, achieving up to 25.61% error reduction in ACCAD with PhysMoP-FT. The relative improvements were pronounced in both short-term and long-term predictions, particularly in PhysMoP at 120 ms. These results further demonstrate the effectiveness of TL even in physics-based approaches for HMP, as the short-term predictions in PhysMoP are mainly driven by the physics block, which solves the Euler-Lagrange equations.

Compared with the MPJPE evaluation, the MAE results are highly consistent and align with the previous discussion. Again, FT proved to be slightly more effective than FAL, but the performance of the two TL methods remains comparable.

3) JAD Evaluation: Table III shows the JAD evaluation. Similarly to MAE, we report results only for the models

TABLE I
MPJPE RESULTS ON ACCAD, GRAB, AND DANCEDB FOR ALL FOUR HMP MODELS, COMPARING TRAINING FROM SCRATCH (SC) WITH THE TWO TL APPROACHES [FINE-TUNING (FT) AND FREEZING AND ADDING NEW LAYERS (FAL)]

	ACCAD Dataset						GRAB Dataset						DanceDB Dataset					
Milliseconds	120	200	400	600	1000	%	120	200	400	600	1000	%	120	200	400	600	1000	%
PVRED-SC	44.5	65.2	85.3	95.2	116.8	-	13.6	22.3	39.1	46.9	57.2	-	58.4	85.0	123.7	140.7	151.9	-
PVRED-FT	**24.9**	**40.7**	**64.1**	**82.2**	94.5	**23.09**	**9.1**	**16.0**	**31.3**	**39.8**	**51.7**	**15.45**	**40.3**	**64.9**	**104.5**	**123.2**	**131.2**	**14.78**
PVRED-FAL	27.0	41.9	66.5	83.4	**92.0**	22.69	11.5	18.1	31.8	40.0	52.8	13.06	48.5	72.6	109.8	129.0	137.9	9.93
HRI-SC	31.7	48.8	75.1	90.9	109.9	-	9.2	16.2	32.0	40.0	52.4	-	48.6	73.3	113.0	131.6	143.2	-
HRI-FT	**22.1**	**34.9**	**52.7**	70.3	90.5	24.00	7.0	12.7	26.5	34.7	46.5	14.99	**34.4**	**55.1**	**89.4**	**108.6**	124.4	**18.56**
HRI-FAL	22.3	35.0	53.1	**69.9**	**88.4**	**24.84**	**6.8**	**12.5**	**25.4**	**33.3**	**45.1**	**17.82**	34.7	55.4	90.3	109.7	**124.2**	17.98
siMLPe-SC	29.3	47.2	73.1	93.1	105.4	-	10.1	17.2	30.8	37.6	47.9	-	38.8	61.8	101.7	124.1	135.7	-
siMLPe-FT	**21.3**	**33.7**	54.1	73.1	97.0	19.02	**6.0**	**11.7**	**25.9**	34.9	46.9	10.13	31.9	52.5	87.8	108.4	122.4	12.20
siMLPe-FAL	21.8	34.4	**54.0**	**71.6**	**94.4**	**20.16**	**6.0**	11.9	26.4	**33.9**	**46.0**	**11.71**	**31.6**	**51.9**	**85.4**	**107.1**	**119.1**	**14.48**
PhysMoP-Sc	4.9	8.3	21.2	40.4	74.3	-	1.0	1.6	4.7	12.1	34.1	-	5.1	10.9	32.5	68.2	128.3	-
PhysMoP-Ft	2.3	3.7	**11.0**	**24.8**	**59.3**	**32.95**	0.7	**1.1**	**3.6**	10.0	**29.5**	**16.41**	2.5	**6.3**	**21.8**	**50.4**	**102.2**	**24.91**
PhysMoP-Fal	**2.0**	**3.6**	11.4	26.2	61.0	30.23	**0.5**	1.5	**3.6**	**9.9**	29.9	15.62	**2.4**	6.7	22.5	50.7	105.1	22.35

Note: In addition to MPJPE, we report the percentage improvement of FT and FAL relative to SC for each HMP model across the entire prediction length. Best results are in **bold**, second-best are underlined.

TABLE II
MAE RESULTS ON ACCAD, GRAB, AND DANCEDB FOR PVRED AND PHYSMOP, COMPARING TRAINING FROM SCRATCH (SC) WITH THE TWO TL APPROACHES [FINE-TUNING (FT) AND FREEZING AND ADDING NEW LAYERS (FAL)]

	ACCAD Dataset						GRAB Dataset						DanceDB Dataset					
Milliseconds	120	200	400	600	1000	%	120	200	400	600	1000	%	120	200	400	600	1000	%
PVRED-SC	0.669	1.014	1.248	1.334	1.634	-	0.503	0.767	1.091	1.277	1.570	-	1.069	1.425	1.658	1.739	2.001	-
PVRED-FT	**0.454**	**0.657**	**1.025**	**1.221**	**1.293**	**18.68**	0.360	**0.556**	**0.845**	**1.074**	**1.398**	**18.02**	**0.956**	**1.335**	1.632	**1.667**	**1.742**	**6.52**
PVRED-FAL	0.481	0.674	**1.025**	1.231	1.328	18.13	**0.282**	0.684	0.849	1.095	1.424	15.09	0.992	1.354	**1.507**	1.744	1.747	6.39
PhysMoP-SC	0.080	0.143	0.306	0.561	1.092	-	0.027	0.045	0.129	0.343	0.841	-	0.084	0.295	0.717	1.074	1.715	-
PhysMoP-FT	0.046	**0.077**	**0.193**	**0.390**	**0.977**	25.61	0.025	**0.042**	**0.112**	**0.294**	**0.790**	**10.76**	0.044	0.257	0.484	0.831	**1.530**	**22.00**
PhysMoP-FAL	**0.045**	0.088	0.202	0.396	0.978	**24.86**	**0.014**	0.045	0.114	0.303	0.792	9.18	**0.039**	**0.109**	**0.359**	**0.826**	1.604	20.59

Note: In addition to MPJPE, we report the percentage improvement of FT and FAL relative to SC for each HMP model across the entire prediction length. Best results are in **bold**, second-best are underlined.

based on an angle representation, namely PVRED and PhysMoP. The results confirm previous observations: TL approaches consistently outperformed scratch training, FT generally performed better than FAL, and the highest gains were observed in ACCAD.

Overall, all three metrics: 1) MPJPE; 2) MAE; and 3) JAD showed great consistency confirming the accuracy increase of TL when evaluating both 3D coordinates and rotation angles. Additionally, FT and FAL achieved significant improvements over the scratch baseline across all experiments, which included different model architectures and target contexts. These results highlight the strong effectiveness and robust generalization of TL for improving context-specific HMP under data scarcity.

B. Impact of Dataset Distance

We now analyze how the distance between motion datasets affects the relative improvements of using TL methods compared with scratch training. Following [57], [58], we train an RNN-based autoencoder using both source and target datasets and use the latent space embedding for computing the Fréchet inception distance (FID) [59], quantifying the datasets distance. To compute the source intra-FID, we randomly split the dataset into two equal size subsets and average the FID results over five random splits. In Table IV, we present the FID results.

Among the three targets, ACCAD showed the lowest FID with the source and the largest TL gain. This is because ACCAD includes generic actions such as walking or running, which are also present in the source. In contrast, GRAB exhibits the highest FID with the source data and the lowest overall TL improvement. This correlation is likely associated with the high specificity of the actions in GRAB, which involve object manipulation with a stationary lower body, limiting the effectiveness of knowledge transfer. In contrast, DanceDB lies between the two for both FID and TL improvements. Although dancing actions are highly specific, they involve whole-body motions that better align with the source domain compared with GRAB.

Overall, the FID metric confirms greater TL improvements when the motion primitives of the source and target datasets exhibit higher similarity. Nevertheless, TL approaches also showed notable improvements in GRAB and DanceDB, thereby demonstrating their robustness and effectiveness in enhancing context-specific predictions under data scarcity and higher domain shift.

TABLE III
JAD RESULTS ON ACCAD, GRAB, AND DANCEDB FOR PVRED AND PHYSMOP, COMPARING TRAINING FROM SCRATCH (SC) WITH THE TWO TL APPROACHES [FINE-TUNING (FT) AND FREEZING AND ADDING NEW LAYERS (FAL)]

	ACCAD Dataset						GRAB Dataset						DanceDB Dataset					
Milliseconds	120	200	400	600	1000	%	120	200	400	600	1000	%	120	200	400	600	1000	%
PVRED-SC	0.848	1.202	1.561	1.766	2.053	–	0.366	0.595	1.014	1.206	1.470	–	1.374	1.933	2.705	2.967	3.192	–
PVRED-FT	**0.593**	**0.894**	**1.259**	**1.549**	1.814	**16.91**	**0.264**	**0.450**	**0.825**	**1.031**	**1.356**	**13.88**	**1.157**	**1.718**	**2.448**	**2.728**	**2.913**	**9.02**
PVRED-FAL	0.615	0.907	1.285	1.568	**1.809**	15.88	0.301	0.488	0.848	1.052	1.388	11.22	1.246	1.806	2.537	2.829	3.035	5.57
PhysMoP-SC	0.133	0.220	0.485	0.838	1.515	-	0.041	0.067	0.163	0.366	0.938	-	0.147	0.311	0.810	1.507	2.659	-
PhysMoP-FT	0.078	**0.129**	**0.315**	**0.624**	1.295	**23.02**	0.037	0.061	**0.149**	**0.331**	0.865	8.91	0.079	0.258	**0.642**	1.244	**2.350**	15.93
PhysMoP-FAL	**0.075**	0.137	0.327	0.627	**1.285**	22.89	**0.016**	**0.049**	0.151	0.335	**0.862**	**9.47**	**0.069**	**0.193**	0.650	**1.238**	2.395	**15.70**

Note: In addition to MPJPE, we report the percentage improvement of FT and FAL relative to SC for each HMP model across the entire prediction length. JAD values are scaled by a factor of 10 to improve readability. Best results are in **bold**, second-best are underlined.

TABLE IV
FID BETWEEN EACH PAIR OF SOURCE-TARGET DATASETS AND THE SOURCE INTRA-FID AS BASELINE

Target Dataset	ACCAD	GRAB	DanceDB	*Source (intra-FID)*
FID	0.216	0.404	0.362	0.172
TL gain (%)	24.62	14.40	16.90	-

Note: Additionally, for each target dataset we report the average MPJPE improvement of TL.

TABLE V
NUMBER OF PARAMETERS FOR EACH MODEL IN THE STANDARD CONFIGURATION AND WITH FAL (ADDING A LINEAR LAYER)

Model	#Params	#Params (FAL)	Model Type	TL Gain (%)	
				FT	FAL
PVRED	10 186 824	10 192 080	RNN	17.77	15.22
HRI	3 203 438	3 208 408	GCN & attention	19.18	20.21
SIMLPE	133 524	136 494	MLP	13.78	15.45
PHYSMOP	2 160 404	2 168 468	MLP & physics	24.76	22.73

Note: The TL gain column shows the average MPJPE percentage improvement of FT and FAL over the scratch baseline, across all target datasets.

C. Impact of Model Architecture

Table V summarizes the total number of parameters, architecture type, and average MPJPE improvement of TL methods across all the experiments. From the analysis of model complexity, the TL efficiency is not strictly related to the number of parameters. For instance, PVRED includes more than 10M parameters, yet its relative improvement is lower than PhysMoP, which has approximately four times fewer parameters. This suggests that TL effectiveness in HMP is more influenced by the network design rather than the model parameter size.

PhysMoP and HRI use a structured and semantically meaningful approach to learn human motion by integrating physics modeling and GCN with motion attention, respectively. Instead, RNNs are primarily designed to capture temporal dependencies within individual sequences. Consequently, PVRED inherits a limited capability to learn generalizable motion across different sequences from its base architecture. However, the results for siMLPe indicate that excessively lightweight architectures may lack the capability of leveraging large-scale source datasets effectively, leading to limited TL improvements.

The differences in the effectiveness of FT and FAL also appear to depend on the network structure. While FT preserves the full architecture, FAL introduces an additional linear layer and freezes the pretrained parameters. This strategy tends to be more beneficial for architectures that encode motion features in a hierarchical structure. From our results, FAL performed better in siMLPe and HRI, both of which employ multiple stacked MLP and GCN layers. Overall, these findings suggest that FT is the more reliable approach, although models with deeper and layered architectures can benefit more from the parameter-freezing regularization introduced by FAL.

D. Pretraining on AMASS Versus H36M

To gain a deeper understanding of how the pretraining dataset impacts TL performance in HMP, we compared our MAE results with those from previous work on FT [11]. In this study, we used AMASS as the source dataset, while in [11] Human 3.6M (H36M) was used instead. We ensured consistency between the two experimental settings by using the same model architecture, prediction time range, and target motion sequences. To address the skeletal differences between H36M and AMASS and highlight the relative improvement of PVRED-FT over PVRED-SC, we normalized the MAE results by their maximum values. In Fig. 4, we show the normalized MAE results for PVRED-SC and PVRED-FT when pretraining on H36M and AMASS. We quantified the relative improvement by computing the area under the curves (AUC) when pretraining on H36M (ACCAD: 1.76, GRAB: 0.68, DanceDB: 0.17) and when pretraining on AMASS (ACCAD: 3.26, GRAB: 3.13, DanceDB: 1.27). Overall, the performance gap was highly superior when using AMASS for pretraining. This dataset is significantly larger than H36M, confirming that using a larger set of motion samples enables the model to learn a stronger prior on human motion.

When using H36M, the TL improvement was substantial only in ACCAD, while the performance of PVRED-SC and PVRED-FT became closer in GRAB and DanceDB. In contrast, our results showed a significant boost across all three target datasets. Compared with H36M, the AMASS dataset includes a more diverse set of action categories, allowing PVRED to better

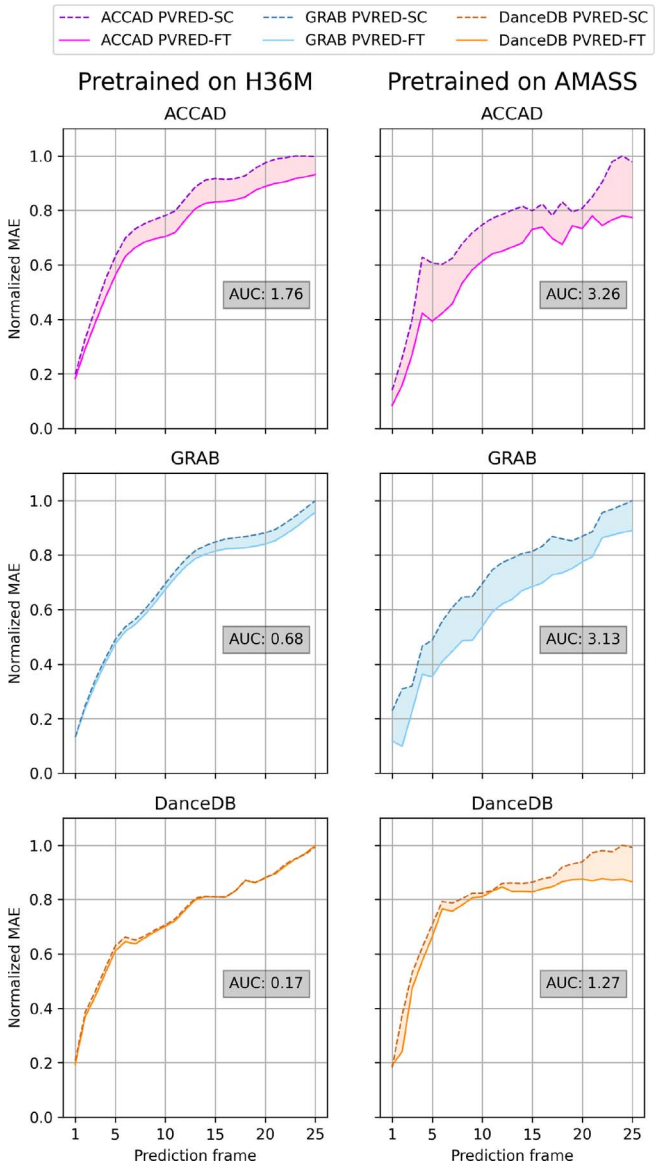

Fig. 4. Evaluation comparison between pretraining on H36M and on AMASS. The improvement of PVRED-FT over PVRED-SC is measured by the AUC, corresponding to the filled areas. Pretraining on AMASS is significantly better.

generalize with stronger domain shift. These results highlight the importance of using a large and diverse dataset for pretraining to enhance the effectiveness of TL across multiple HMP contexts.

E. TL Evaluation

In this section, we present and discuss the TL metrics results, which are shown in Table VI. These metrics evaluate the training performance of the tested approaches, highlighting the training efficiency of TL methods compared with the scratch baseline.

1) JSP Evaluation: The JSP measures the contribution of pretrained knowledge by comparing the initial training loss. FT consistently showed positive values, achieving up to 80.8% increase using siMLPe on GRAB. These results highlight the contribution of pretrained knowledge in HMP, proving its effectiveness in model training. In contrast, loss improvement in FAL was limited because most model parameters are initially frozen, which limits the learning capacity. This was particularly evident in PVRED, where the autoregressive structure of the RNN was strongly constrained. On the other hand, siMLPe and PhysMoPs were less affected by parameter freezing, as both FT and FAL achieved similar JSP values.

TABLE VI
TL METRICS RESULTS FOR ALL FOUR HMP MODELS TRAINED ON ACCAD, GRAB, AND DANCEDB

	ACCAD Dataset			GRAB Dataset			DanceDB Dataset		
	JSP	MEP	TTC	JSP	MEP	TTC	JSP	MEP	TTC
PVRED-FT	**67.4**	**15.0**	**96.3**	**78.7**	**12.1**	**97.8**	**51.4**	**7.4**	**98.8**
PVRED-FAL	–12.0	13.2	86.3	–25.7	10.9	89.0	–12.9	4.7	61.2
HRI-FT	**47.3**	**31.0**	17.2	**10.9**	13.0	76.7	**25.2**	**16.4**	**90.3**
HRI-FAL	1.0	29.2	**30.5**	–1.7	**13.7**	**80.4**	0.8	15.6	50.5
siMLPe-FT	**52.0**	17.4	**96.7**	**80.8**	**12.0**	**98.7**	60.6	5.7	**99.8**
siMLPe-FAL	48.3	**19.0**	94.7	80.2	11.1	97.0	**61.7**	**7.4**	98.1
PhysMoP-FT	59.2	**37.0**	**96.1**	64.4	**22.7**	98.6	60.4	32.1	**98.1**
PhysMoP-FAL	**60.1**	36.7	94.5	**66.5**	22.5	**98.8**	**62.4**	**32.4**	97.8

Note: The metrics report the percentage improvement over the corresponding scratch training. Best results are in **bold**.

Overall, the JSP results confirm the contribution of pretrained knowledge while emphasizing the need to adapt the learned motion primitives to the specific target context.

2) MEP Evaluation: The MEP results support the quantitative findings discussed in Section V-A, showing that FT and FAL consistently reduced the training loss compared with scratch models. Interestingly, in most cases, the percentage improvement in MEP was lower than that observed in the accuracy results. This indicates that TL approaches tend to perform better over scratch during testing than during training, highlighting the superior generalization capabilities of FT and FAL. Pretraining on the source dataset provides a more comprehensive knowledge of the human body, which helps the model generalize to previously unseen motion sequences.

3) TTC Evaluation: The TTC measures the convergence speed of TL approaches compared with the scratch baseline. Both FT and FAL converged significantly faster across all target datasets and models. Overall, FT was the fastest approach, often converging more than 95% quicker than scratch. Notably, FAL also demonstrated substantial training speed gains, despite the initial freezing of model parameters. The TTC results highlight another key advantage of using TL in HMP: it enables rapid adaptation of the prediction model to specific target actions. As the pretrained model is shared across all target domains, a single pretrained checkpoint can be used for multiple context-specific applications. This enables the rapid deployment of the HMP model real-world application, reducing computational costs and training time.

F. Qualitative Evaluation

In Fig. 5, we show example predictions from one of the TL approaches (FT) and the scratch baseline. Overall, the predicted

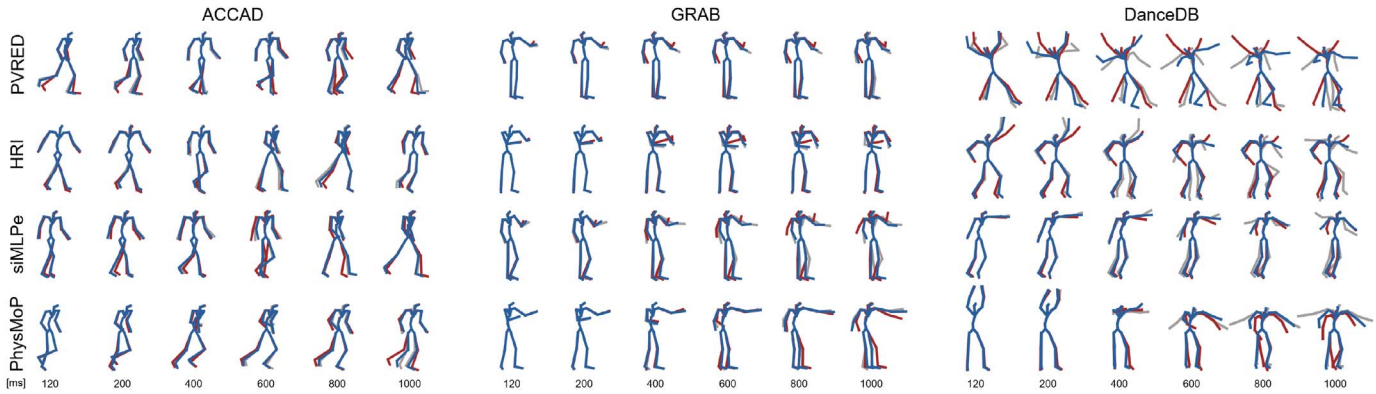

Fig. 5. Visual examples from FT (blue) and the scratch baseline (red) compared with the ground truth (gray).

sequences from FT are consistently closer to the ground truth compared with scratch training. This is particularly evident in ACCAD-FT, where the model maintains high precision also in long-term predictions. The sequences from DanceDB are instead more challenging to predict due to rapid, full-body movements. Here, both FT and scratch models struggle to make accurate predictions, although FT outputs are more precise. GRAB, on the other hand, involves only upper body motion, which reduces the overall prediction error, consistent with the trends observed in the quantitative evaluation.

VI. Conclusion

This article presents a unified framework and analytical study of TL effectiveness in HMP to improve prediction accuracy under data scarcity. To ensure the generalizability of our findings, the experiments included multiple HMP models, TL approaches, and target datasets. Our results consistently show that TL improves prediction accuracy across all the tested models and target contexts compared with scratch training. Additionally, TL methods achieved faster convergence, significantly reducing training time. Overall, these findings demonstrate that TL has strong potential to improve HMP accuracy, enhancing the applicability of these systems for real-world tasks where context-specific data is often scarce. Furthermore, the results characterize the roles of dataset similarity and model architecture, highlighting the importance of using large and diverse source datasets, in combination with HMP models capable of learning structured motion features. In future work, we plan to integrate TL methods in an HMP application to better understand how they can improve motion predictions in real-world contexts to further increase HMP applicability.

References

[1] M. Gulzar, Y. Muhammad, and N. Muhammad, "A survey on motion prediction of pedestrians and vehicles for autonomous driving," *IEEE Access*, vol. 9, pp. 137957–137969, 2021.

[2] J.-L. Ren, Y.-H. Chien, E.-Y. Chia, L.-C. Fu, and J.-S. Lai, "Deep learning based motion prediction for exoskeleton robot control in upper limb rehabilitation," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, 2019, pp. 5076–5082.

[3] A. Rudenko, L. Palmieri, M. Herman, K. M. Kitani, D. M. Gavrila, and K. O. Arras, "Human motion trajectory prediction: A survey," *Int. J. Robot. Res.*, vol. 39, no. 8, pp. 895–935, 2020.

[4] M.-L. Lee, W. Liu, S. Behdad, X. Liang, and M. Zheng, "Robot-assisted disassembly sequence planning with real-time human motion prediction," *IEEE Trans. Syst., Man, Cybern.: Syst.*, vol. 53, no. 1, pp. 438–450, Jan. 2023.

[5] J. Martinez, M. J. Black, and J. Romero, "On human motion prediction using recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2891–2900.

[6] K. Lyu, H. Chen, Z. Liu, B. Zhang, and R. Wang, "3D human motion prediction: A survey," *Neurocomputing*, vol. 489, pp. 345–365, Jun. 2022.

[7] L.-Y. Gui, Y.-X. Wang, D. Ramanan, and J. M. Moura, "Few-shot human motion prediction via meta-learning," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 432–450.

[8] Z. Cai et al., "Playing for 3D human recovery," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 12, pp. 10533–10545, Dec. 2024.

[9] G. Barquero, S. Escalera, and C. Palmero, "BeLFusion: Latent diffusion for behavior-driven human motion prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2023, pp. 2317–2327.

[10] F. Zhuang et al., "A comprehensive survey on transfer learning," *Proc. IEEE*, vol. 109, no. 1, pp. 43–76, 2020.

[11] M. Casarin, M. Vanuzzo, M. Guidolin, M. Reggiani, and S. Michieletto, "Enhancing robot collaboration by improving human motion prediction through fine-tuning," in *Proc. IEEE Int. Conf. Automat. Sci. Eng. (CASE)*, 2024, pp. 3358–3364.

[12] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black, "AMASS: Archive of motion capture as surface shapes," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2019, pp. 5442–5451.

[13] A. M. Lehrmann, P. V. Gehler, and S. Nowozin, "Efficient nonlinear Markov models for human motion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2014, pp. 1314–1321.

[14] M. Brand and A. Hertzmann, "Style machines," in *Proc. ACM Int. Conf. Comput. Graph. InterAct. Techn. (SIGGRAPH)*, Jul. 2000, pp. 183–192.

[15] R. Urtasun, D. J. Fleet, A. Geiger, J. Popović, T. J. Darrell, and N. D. Lawrence, "Topologically-constrained latent variable models," in *Proc. Int. Conf. Mach. Learn (ICML)*, 2008, pp. 1080–1087.

[16] J. M. Wang, D. J. Fleet, and A. Hertzmann, "Gaussian process dynamical models for human motion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 283–298, Feb. 2008.

[17] G. W. Taylor, G. E. Hinton, and S. Roweis, "Modeling human motion using binary latent variables," in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, pp. 1345–1352.

[18] W. Mao, M. Liu, M. Salzmann, and H. Li, "Learning trajectory dependencies for human motion prediction," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9489–9497.

[19] K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik, "Recurrent network models for human dynamics," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2015, pp. 4346–4354.

[20] D. Pavllo, D. Grangier, and M. Auli, "QuaterNet: A Quaternion-based recurrent model for human motion," 2018, *arXiv:1805.06485*.

[21] H. Wang, J. Dong, B. Cheng, and J. Feng, "PVRED: A position-velocity recurrent encoder-decoder for human motion prediction," *IEEE Trans. Image Process*, vol. 30, pp. 6096–6106, 2021.

[22] W. Mao, M. Liu, and M. Salzmann, "History repeats itself: Human motion prediction via motion attention," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Aug. 2020, pp. 474–489.

[23] M. Li, S. Chen, Y. Zhao, Y. Zhang, Y. Wang, and Q. Tian, "Dynamic multiscale graph neural networks for 3D skeleton based human motion prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 214–223.
[24] L. Dang, Y. Nie, C. Long, Q. Zhang, and G. Li, "MSR-GCN: Multi-scale residual graph convolution networks for human motion prediction," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2021, pp. 11447–11456.
[25] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, and Q. Tian, "Symbiotic graph neural networks for 3D skeleton-based human action recognition and motion prediction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 6, pp. 3316–3333, Jun. 2022.
[26] S. Yang, H. Li, C.-M. Pun, C. Du, and H. Gao, "Adaptive spatial-temporal graph-mixer for human motion prediction," *IEEE Signal Process. Lett.*, vol. 31, pp. 1244–1248, 2024.
[27] A. Bouazizi, A. Holzbock, U. Kressel, K. Dietmayer, and V. Belagiannis, "MotionMixer: MLP-based 3D human body pose forecasting," 2022, *arXiv:2207.00499*.
[28] W. Guo, Y. Du, X. Shen, V. Lepetit, X. Alameda-Pineda, and F. Moreno-Noguer, "Back to MLP: A simple baseline for human motion prediction," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. Workshops (WACVW)*, 2023, pp. 4809–4819.
[29] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 7132–7141.
[30] T. Maeda and N. Ukita, "MotionAug: Augmentation with physical correction for human motion prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 6427–6436.
[31] Z. Zhang, Y. Zhu, R. Rai, and D. Doermann, "PimNet: Physics-infused neural network for human motion prediction," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 8949–8955, Oct. 2022.
[32] Y. Zhang, J. O. Kephart, and Q. Ji, "Incorporating physics principles for precise human motion prediction," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. Workshops (WACVW)*, 2024, pp. 6164–6174.
[33] G. Gómez-Izquierdo, J. Laplaza, A. Sanfeliu, and A. Garrell, "Enhancing context-aware human motion prediction for efficient robot handovers," 2025, *arXiv:2503.00576*.
[34] X. Liang, L. Sheng, J. Cai, S. Li, and Y. Shi, "Context-aware deep learning model for 3D human motion prediction in human-robot collaborative construction," in *Proc. Comput. Civil Eng.*, 2024, pp. 479–487.
[35] H.-S. Moon and J. Seo, "Fast user adaptation for human motion prediction in physical human–robot interaction," *IEEE Robot. Autom. Lett.*, vol. 7, no. 1, pp. 120–127, Jan. 2021.
[36] C. Zang, M. Pei, and Y. Kong, "Few-shot human motion prediction via learning novel motion dynamics," in *Proc. 29th Int. Joint Conf. Artif. Intell.*, 2021, pp. 846–852.
[37] M. Iman, H. R. Arabnia, and K. Rasheed, "A review of deep transfer learning and recent advancements," *Technologies*, vol. 11, no. 2, p. 40, 2023.
[38] C. Doersch and A. Zisserman, "Sim2real transfer learning for 3D human pose estimation: Motion to the rescue," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, vol. 32, 2019.
[39] L. Zhang, D. Soselia, R. Wang, and E. M. Gutierrez-Farewik, "Lower-limb joint torque prediction using LSTM neural networks and transfer learning," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 600–609, 2022.
[40] L. Ballan, F. Castaldo, A. Alahi, F. Palmieri, and S. Savarese, "Knowledge transfer for scene-specific motion prediction," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 697–713.
[41] R. Akabane and Y. Kato, "Pedestrian trajectory prediction based on transfer learning for human-following mobile robots," *IEEE Access*, vol. 9, pp. 126172–126185, 2021.
[42] E. Aksan, M. Kaufmann, P. Cao, and O. Hilliges, "A spatio-temporal transformer for 3D human motion prediction," in *Proc. IEEE Int. Conf. 3D Vis. (3DV)*, 2021, pp. 565–574.
[43] Z. Zhu, K. Lin, A. K. Jain, and J. Zhou, "Transfer learning in deep reinforcement learning: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 11, pp. 13344–13362, Nov. 2023.
[44] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "SMPL: A skinned multi-person linear model," *ACM Trans. Graph*, vol. 34, pp. 1–16, Oct. 2015.
[45] S. Ghorbani et al., "MoVi: A large multi-purpose human motion and video dataset," *PLoS One*, vol. 16, no. 6, 2021, Art. no. e0253157.
[46] N. F. Troje, "Decomposing biological motion: A framework for analysis and synthesis of human gait patterns," *J. Vis.*, vol. 2, no. 5, pp. 371–387, 2002.
[47] Mocapdata. EyesJapanDataset. [Online]. Available: http://mocapdata.com/
[48] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber, "Mocap Database HDM05," *Institut Für Informatik II, Universität Bonn*, vol. 2, p. 7, Jun. 2007.
[49] M. Loper, N. Mahmood, and M. J. Black, "MoSh: Motion and shape capture from sparse markers," *ACM Trans. Graph*, vol. 33, no. 6, pp. 1–13, 2014.
[50] M. Trumble, A. Gilbert, C. Malleson, A. Hilton, and J. Collomosse, "Total capture: 3D human pose estimation fusing video and inertial sensors," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2017, pp. 1–13.
[51] CMU Graphics Lab. CMU graphics lab motion capture database. [Online]. Available: http://mocap.cs.cmu.edu/
[52] C. Mandery, O. Terlemez, M. Do, N. Vahrenkamp, and T. Asfour, "Unifying representations and large-scale whole-body motion databases for studying human motion," *IEEE Trans. Robot*, vol. 32, no. 4, pp. 796–809, Aug. 2016.
[53] KangKang Yin Goh Jing Ying, "SFU motion capture database." [Online]. Available: https://mocap.cs.sfu.ca/
[54] O. Taheri, N. Ghorbani, M. J. Black, and D. Tzionas, "GRAB: A dataset of whole-body human grasping of objects," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 581–600.
[55] Computer Graphics Lab, University of Cyprus. DanceDB. [Online]. Available: https://dancedb.eu/
[56] Advanced Computing Center for the Arts and Design. ACCAD. [Online]. Available: https://accad.osu.edu/research/motion-lab/mocap-system-and-data
[57] C. Guo et al., "Action2motion: Conditioned generation of 3D human motions," in *Proc. ACM Int. Conf. Multimedia*, 2020, pp. 2021–2029.
[58] H. Yi, J. Thies, M. J. Black, X. B. Peng, and D. Rempe, "Generating human interaction motions in scenes with text control," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2024, pp. 246–263.
[59] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local Nash equilibrium," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 6626–6637.

Open Access funding provided by 'Universitã degli Studi di Padova 2025–2027 Deposit Account' within the CRUI CARE Agreement