2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA) | 978-1-6654-6283-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICMLA55696.2022.00177

PerMTL: A Multi-Task Learning Framework for Skilled Human Performance Assessment

Indrajeet Ghosh[1,2], Avijoy Chakma[1], Sreenivasan Ramasamy Ramamurthy[3], Nirmalya Roy[1,2], Nicholas Waytowich[4]
[1]Department of Information Systems, University of Maryland Baltimore County
[2]Center for Real-time Distributed Sensing and Autonomy (CARDS), University of Maryland Baltimore County
[3]Department of Computer Science, Bowie State University, [4]DEVCOM, Army Research Laboratory
United States
[1,2]{*indrajeetghosh, achakma1, nroy*}*@umbc.edu*, [3]*sramamurthy@bowiestate.edu*, [4]*nicholas.r.waytowich.civ@army.mil*

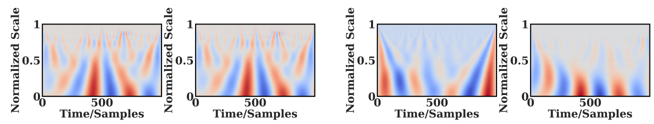
***Abstract*—Intelligent and complex human motion analysis can help design the next generation IoT and AR/VR systems for automated human performance assessment. Such an automated system can help advocate the interpretability and translatability of complex human motions, intelligent motion feedback, and fine-grained motion skill assessment to design next-generation interactive human-machine teaming systems. Motivated by this, we design a wearable sensing framework for assessing the players' performance and consider a live *badminton game* as our use case. Generally, the players on the field try to improve their performance by focusing on fast and synchronous coordination of their limbs' reflex actions to have the ideal body postures to perform the desired shot. Learning the minute dissimilarities and distinctive traits from each limb of the players simultaneously can help assess the players' performance and specific skillsets during a game. This paper proposes a multi-task learning framework, *PerMTL* to learn the shared features from each player's limb. The *PerMTL* comprises a task-specific regressor output layer that helps to determine the dissimilarities and distinctive traits between the player's limbs for collective inference in a body sensor network (BSN) environment. We evaluate the *PerMTL* framework using publicly available Badminton Activity Recognition (BAR) and Daily and Sports Activities (DSA) datasets. Empirical results indicate that *PerMTL* achieves R^2 Score of $\approx$ *82%* in predicting the players' performance.**

***Index Terms*—Multi-task Learning, Sports Analytics, Activity Recognition, Error Estimation, Performance Assessment**

I. Introduction

Wearable smart devices are becoming an integral part of our daily activities. Such wearable devices are widely used in developing applications in smart homes [1], gait analysis [2], and sports [3]. The fusion of various sensors such as motion, temperature and heart rate sensors provide ample opportunities to discover the activity execution pattern and enable us to investigate the prospects for future improvement. Given the scenario, sports analytics is no different in benefiting from the commercial adaptation of smart wearables during the practice session accumulating player statistics, player profiling and performance improvement [4]. Nonetheless, commercial device's technical development mainly focuses on gathering player statistics and provides less explorative analysis and feedback to the players. In our research, we work towards filling this gap.

Success in competitive sports depends on several factors, such as a player's physical fitness, technical skillset, and the adaptation of strategies at various crucial moments. This paper focuses on developing a framework that supports technical skill improvement in the Badminton sport. ***Badminton*** is a racquet game played in singles and doubles format across the globe and a part of Olympic games. In badminton, players need to perform acrobatic activities that highly involve synchronous hands and legs maneuvering in the ground and the air. Effective and efficient maneuvering skill requires hours of training and coaching. The difference in maneuvering skill significantly affects the player's performance.



(a) Professional Player (b) Novice Player

Fig. 1: Magnitude scalograms of dominant limbs for *Smash Overhead Forehand Shot*

The subtle difference in agility and skill can be captured and assessed to provide valuable feedback to the comparatively less skilled player (trainee/novice). We hypothesize that each player's limb is unique and distinctive. Estimating such distinctive traits will improve the players' performance irrespective of factors (height, dominance limb, weight, etc.) in the game. To support our proposition, Figures 1a and 1b demonstrate the magnitude of scalograms corresponding to the ***Smash overhead forehand shots*** from the professional and novice players, respectively. Magnitude values were calculated from the 3-axis accelerometer, magnetometer and gyroscope values. The scalogram suggests that the professional player performs a defined repetitive pattern that might contribute to low variations and sparse data points. Furthermore, the scalogram from the professional player in Fig. 1a also reveals a steady signal power at the same frequency band throughout the time segment to complement this inference. Whereas, in Fig. 1b, such definitive repetitive patterns are not visible in the case of the novice player. Fewer repetitive patterns suggest a higher error occurrence in the novice player than the professional player. Figure 1a and 1b suggests that the IMU data can capture the differences between the professional

978-1-6654-6283-9/22/$31.00 ©2022 IEEE
DOI 10.1109/ICMLA55696.2022.00177

player and the novice player proficiency.

Fig. 2a depicts the broad overview of the ***PerMTL*** framework that aims to determine the limb maneuvering quality. Towards achieving the goal, it processes the collected IMU data from four body positions where each IMU sensor is integrated with accelerometry, gyroscope and magnetometer. In addition, the error propagation module helps us to measure the players' minute dissimilarities and distinctive traits within the framework. Furthermore, the inference module provides a collective performance assessment and resource consumption profiling of the framework. The motivation of profiling of the *PerMTL* framework is to exemplify the required inference time compared to the state-of-the-art proposed algorithms.

Key contributions of the paper are summarized as follows:

- **Generalized Multi-task Learning Framework**- We postulate a novel approach to learning by employing a multi-task learning, ***PerMTL*** framework to learn the shared features from each limb of the players through shared layers. It is followed by task-specific output regression layers, where the difference between each limb is retained.
- **Evaluation**- We evaluate *PerMTL* framework using publicly available Badminton Activity Recognition (BAR) and Daily and Sports Activities (DSA) datasets (comprises of activities of daily living (ADLs), instrumental activities of daily living (IADLs) and sports activities). We introduce a ***cross-person validation strategy*** to assess the scalability of the *PerMTL* framework. We observe that *PerMTL* achieves R^2 Score of $\approx$ ***82%*** for predicting the players' performance.
- **Profiling**- We benchmark the *PerMTL* framework and report the hardware resource consumption (time) of the PyTorch operations. We propose ***layer-wise time complexity*** profiling of the ***PerMTL*** framework to demonstrate the reliability and robustness for the real-time inference.

II. Related Work

This section summarises the related work applied methodologies into three categories: IoT wearables in sports analytics, multi-task learning for activity recognition and performance assessment. We mainly focus on the aspect of our approach that is different from the existing approaches.

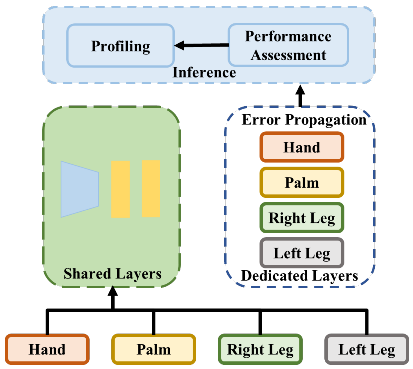
IoT wearables are making swift progress in developing various application and research works in different sports such as tennis [5], soccer [3], badminton [6], etc. This literature study mainly discusses the research works focusing on Badminton as discussing other works is beyond this paper's scope. In [6], the authors attempted to classify nine badminton shots and the associated body movements by leveraging Convolution Neural Networks (CNNs). The authors placed sensors on three body positions(the wrist, bottom of the racket's grip and the upper arm) and collected badminton data from two right-handed players in the data collection. However, the work shortfall focuses on recognizing a limited number of badminton strokes using only upper limb movements data. In contrast, the lower limb movements data did not consider in their study. Prior additional research works mainly focused on analyzing the smash shot [7], [8], where the authors proposed approaches to measure the acceleration and the movement of the upper and lower arm of the player. All these literature works focus on classifying and recognizing various shots, and the feedback mechanism to the beginner-level player is still missing. We aim to extend the existing research direction from the classification task to develop a feedback-providing mechanism so that any entry-level player can be informed of the mistakes, make the correction, and improve their performance.

Multi-task learning [9] is an inductive transfer learning methodology that accomplishes to increases the generalization performance of a proposed approach by exploiting the commonalities and differences among various tasks. In [10] leverages a multi-task long short-term memory (LSTM) model to accomplish two tasks - accurately classify the activity types and estimate the intensity of each activity. Both tasks play a substantial role in healthcare applications, such as fitness tracking and patient monitoring. Furthermore, in [11], the authors investigated recognizing collective ADLs and sports activities using a single model. The authors experimented with several state-of-the-art approaches and found that multi-task variants result in increased performance. Further, the authors collected the C-Sports dataset, which comprised eleven different sports with five different activities and used the evaluation dataset. In our proposed work, we aim to borrow the benefit of using multi-task learning to model to formulate an end to end data-driven pipeline to estimate the difference between the distinctive traits of the players' limbs.

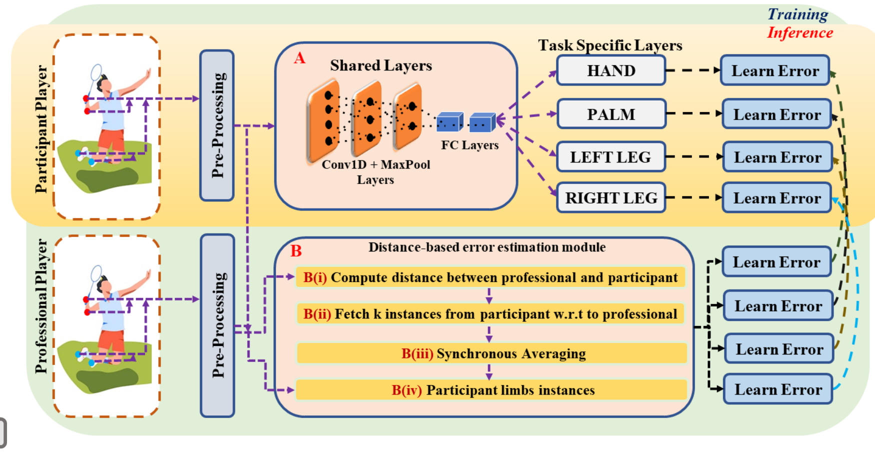
Performance Assessment is an essential aspect of activity detection for long-term monitoring and training purposes. However, such assessments are often carried out by coaches who depend on their previous experiences, and such judgments are prone to subjectivity. AAC [12] considered the wearables that perform mutually to identify not only activities but also to evaluate them qualitatively using the data of several sensor nodes attached to the body. It also provide detailed feedback for the improvement of the execution. AAC focuses on the online assessment of periodic human activity within a wireless body area network. To the best of our knowledge, not too many literary works have concentrated on the players' performance assessment. Similar to [13], ***PerMTL*** also uses multiple body-worn IMU sensors focused on assessing sports activity. However, it differs in that the intuitive assessment is accomplished through a novel multi-tasking approach that operates end-to-end.

III. Preliminary State-of-the-art Study

In the state-of-the art studies [14], [15], propose an instance-based template matching scoring algorithm, a distance-based error learning *(DBEL)*. It enables to capture and detect the minute discrepancies and distinctive traits of the lower limbs (both legs) between the professional players and other players from different levels of expertise. The objective was to estimate the players' performance using a body sensor network environment. The studies denote upper limbs movements as ***stroke*** and lower limbs as ***stance***. ***Error*** is defined as the



(a) Overview of *PerMTL* Framework



(b) Overall Architecture: **(A)** *Multi-task learning module*: An end-to-end data-driven error estimation pipeline to study and capture the distinctive traits between each player's limb simultaneously. **(B)** *Distance-based error learning (DBEL) module* to determine the closest k number of data instance (instance-based matching learning) approach to learn the ideal body posture from other players w.r.t the professional players.

minute discrepancies between the stance & stroke performed by the professional w.r.t the other players. In a badminton shot, a professional player's data samples might incur some data variance which is trivial to understand. Furthermore, to quantify the inconsistency and variations in the movements of the players' limbs w.r.t the professional players, they computed the Euclidean distance of lower limb samples between the professional and other players. However, a few state-of-the-art preliminary studies' limitations motivated us to develop a multi-task learning framework that can overcome the challenges. Furthermore, we highlighted a few of the challenges of the preliminary studies along with relevant solutions below:

A. Challenges

- ***Computation Resources Requirement***: One of the major challenges of the preliminary studies is computation resources requirement. To showcase the challenges, we perform ***components-wise time*** profiling of the scoring framework, *DBEL* module[1] highlighted in Table I. We notice that the components B(i) and B(ii) are the performance bottleneck of the *DBEL* overall pipeline as it took $\approx$ 27 minutes and consumed 60% of 64 GB RAM of memory in total to run an experiment. Due to high computation resources requirements, deploying the *DBEL* module in resource-constrained devices is not feasible.
- ***Technical Limitations***: *DBEL* framework lacks in generalizability and scalability characteristics because of the reliance on the professional datasets'. It also employs handcrafted error metrics (MSE, $RMSE$, $MdAE$, etc.) to compute the players' error. Moreover, the handcrafted error calculation triggers a bottleneck effect for the real-time deployment environment due to the high computational cost.
- ***Limited Data at Inference Time***: Another challenge is the limited influx of data at inference time of the *DBEL* framework because it only considers the lower limb sensory data to infer and estimate the players' performance in the game.

[1] https://github.com/indrajeetghosh/DeCoach

TABLE I: *Component-wise time complexity* profiling of the *DBEL* module

Components	Time
B(i)	$\approx 13.78\ minutes$
B(ii)	$\approx 7.5\ minutes$
B(iii)	$\approx 4.2\ minutes$
B(iv)	$\approx 1.58\ minutes$
B (Total)	$\approx 27\ minutes$

B. Solution

- ***BSN pipeline for collective data inference***: We postulate that multiple incoming sensory data with collective inference will help to build a robust and generalized error estimation framework.
- ***Deep Learning (DL) based Error Estimation Propagation***: We believe that an end-to-end deep learning architecture might help to mitigate and tackle the technical limitations. DL architectures have a higher convergence rate than any shallow learning algorithms due to their ability to learn the feature representation in the latent space. Therefore, *Multi-task Learning* (MTL) architecture gives an advantage where the distinctive traits in an inflow of multiple sources sensory data.
- ***Limb-wise error estimation***: Lastly, to mitigate the modularity challenge, we adopt a limb-wise error estimation module, where we compute the error and capture the distinctive traits for each limb of the players, respectively. Such modularity helps to estimate and predict the most-error prone limb of the players.

IV. System Design

We discusses the problem formulation and proposed framework adopted to tackle the above-discussed challenges.

A. Problem Formulation

Multi-task learning is an inductive transfer learning approach designed to learn multiple tasks simultaneously while exploring the feature representations corresponding to the different task similarities and dissimilarities. Given a dataset, D consists of N denote the number of samples, $(x_i^N, y_i^N)_{i=1}^N$ where x_i and y_i represent the data sample and corresponding label, respectively. The dataset, D, can be considered a group of n sub-datasets without overlapping them, $D_1, ..., D_n$, where the sub-datasets are leveraged to learn different tasks, $T_1, ...T_n$ respectively. Equation 1, is the conventional mathematical formulation of ***MTL*** algorithm [16], [17] where z_i^N corresponds to the weight vector (regression parameters) for the N^{th} task where it maps the x^N sample to corresponding y^N label and Z denote concatenating all the weight vectors $[z_1, z_2, ..., z_n]$. Regularizer $Reg(Z)$ denotes the regularisation of constraints for Z w.r.t the prior knowledge of the data and different hypotheses of the relationship among tasks. Additionally, β is the regularization parameter that stabilizes and balances between the regularizer $Reg(Z)$ and loss function (first part) shown in equation 1.

$$\min_{z=z_1,z_2..z_n} \sum_{i=1}^{N} \mathcal{L}(x_i^N, y_i^N, z_i^N) \ + \beta Reg(Z) \tag{1}$$

We aim to measure the errors from each limb of the participants in a collective inference environment. Therefore, we formulate the error measurements at different limbs as the different tasks of the proposed multi-task learning framework. To accomplish our fundamental goal, we first classify the strokes and corresponding stances by adopting the same classification module reported in the papers [14]. Secondly, we leveraged the computed handcrafted error metrics matrix employed in the state-of-the-art; ***DeCoach*** [14] work obtained from the ***DBEL*** module shown in figure 2b.

B. Proposed Architecture

The overall architecture is depicted in Figure 2b. The architecture is built upon two majors components: MTL module and error propagation module. We describe both components below in detail:

*1) **Multi-task Learning Module**:* The multi-task learning module consists of shared and dedicated layers. Shared layers consists of CNNs layers, max-pooling layers, batch normalization and fully connected layers shared, and responsible for learning the common feature representation among different tasks. We experimented with one-dimensional CNNs layers followed by the max-pooling and batch normalization layers. The extracted features are further processed by the dedicated layers, where the dedicated layers serve as a task-specific regressor unit. We employ four dedicated layers to learn errors from four body sensor position. Initially, the preprocessed limb data is fed through the share layers and the error measurement module. Then, the final feature representation from the shared layer is forwarded and processed by corresponding dedicated layers, which are further processed with the error measurement module. Only the shared layers are updated simultaneously during the training, whereas each task-specific output layer updates independently.

$$EM = \sum_{i=1,j=1}^{n} [E_H^n, E_P^n, E_{RL}^n, E_{LL}^n] \tag{2}$$

Equation 2, EM represents the error matrix obtained by Algorithm 1 for each limb of the participant, where E_H, E_P, E_{RL}, E_{LL} corresponds to the sum of handcrafted error metrics for Hand, Palm, Left Leg and Right Leg, respectively and where n denotes to number of computed error metrics.

Algorithm 1: *Distance-based error learning (DBEL)* module, where the P$_X$ and Q$_{\overline{X}}$ refers to the professional's and participant's windowed data for each limb respectively whereas P$_y$ and Q$_{\overline{y}}$ refers to the professional's and participant's activity labels of each limb data respectively

INPUT: Acquire data from each limb of P_X and $Q_{\overline{X}}$
OUTPUT: Handcrafted error matrix for each limb
1: for each label ∈ P$_y$ ∩ Q$_{\overline{y}}$
2: Extract data: P_X *and* $Q_{\overline{X}}$
3: Compute euclidean distance ← Sample K (25) instances from (P_X and Q$_{\overline{X}}$ ∈ [label = (P$_y$ ∩ Q$_{\overline{y}}$)])
4: P_{X-avg} ← Average of k closest sample instances
5: Error$_{limb}$ ← $append$ [$Compute\ Error$(Q$_{\overline{X}}$, P$_{X-avg}$)]
6: end for
7: return $Error_{limb}$

*2) **Error Propagation in MTL Module**:* The phenomenon of multi-task learning is to learn the feature representation of the tasks simultaneously and predict or classify the tasks accordingly. We leverage the computed handcrafted error scores from the *DBEL* module as ***ground truth*** for the *MTL* module show in Equation 2. Algorithm 2 represents the multi-task error learning module.

Equation 3 denotes the optimization of regressor output function for each limb, where subscript *limb* corresponds to *Hand, Palm, Left and Right Legs*, where N represent the number of data instances. The ***LogCosh loss*** is calculated independently for each limb. The logCosh loss[2] is similar to the mean squared error (MSE), but the difference is that it does not get affected when occasionally inaccurate prediction. The objective of the loss function is to minimize the Loss for each limb of the players shown in equation 3 where X^{actual} and X^{predicted} are values obtained from the ***DBEL*** and ***MTL*** modules, respectively.

$$\mathcal{L}_{Limb} = \frac{1}{N} \sum_{i=1}^{N} \log(\cosh(X_i^{actual} - X_i^{predicted})) \tag{3}$$

V. Experimentation Pipeline

The experiments were conducted on a Linux server. The server housed an Intel i7-6850K CPU, 4x NVIDIA GeForce GTX 1080Ti GPUs and 64 GB RAM. All the codes for

[2]https://github.com/tuantle/regression-losses-pytorch

Algorithm 2: ***Multi-task learning (MTL)*** module where shared weights = *W*, learning rate = *lr*, number of epochs = *E*, update weight = ΔW

INPUT: $Q_{\overline{X}}$ and Error_{limb} matrix
OUTPUT: E_H, E_P, E_{RL}, E_{LL} represent limb-wise error
1 Initialize the MTL pipeline
2: for epoch = 1 to E, total epochs do:
\# *Forward Propogation*
3: for each *limb*:
4: E_H, E_P, E_{RL}, $E_{LL} = \mathcal{L}_{Limb}$
\# *Backward Propogation*
5: ΔW= *lr** $\frac{\partial LE_H}{\partial W}$, *lr** $\frac{\partial LE_P}{\partial W}$, *lr** $\frac{\partial LE_{RL}}{\partial W}$ & *lr** $\frac{\partial LE_{LL}}{\partial W}$
6: end for
7: Until reach the number of epochs *E*
8: return E_H, E_P, E_{RL}, E_{LL}

data preprocessing and deep learning mechanisms were implemented with python. Especially for deep learning tasks were implemented using PyTorch libraries.

A. BAR Dataset Description

The ***BAR*** dataset [14] was collected to studies the participants' performance w.r.t the professional player and also study the relationship between the limbs (hand, palm and both legs). The dataset acquired from a population of 11 participants (8 males and 3 females, average age: 26 years) collected 30 iterations of each of the 12 strokes. The *BAR* dataset was collected in two different scenarios: ***controlled setting*** (where the participants were selected based on their expertise and collected in an indoor sports premises) and ***uncontrolled setting*** (where the participants were randomly selected and collected in an open playground). They employed four Shimmer3 IMU Unit wearable devices. Each IMU sensor was equipped with a three-axis low noise accelerometer [$\pm 2g$], wide range/high noise accelerometer [upto $\pm 16g$], gyroscope, and magnetometer sensors. They placed the four devices on the participant's dominant wrist, dominant palm, left leg and right leg to capture the movements for each stroke. The data were collected based on the participants' expertise performance in the game, which is listed in tables II & III. They recorded the data collection session using an action camera to validate the activity labelling. The recorded videos of the sessions were used as the ground truth to assign the labels to the activities. The dataset consists of two labels: 1) Activity Label, 2) Score Label for each stroke played. The score varies from [0 to 4] depending on how well the participant played the shot [14]. Further, two annotators were assigned to annotate the *BAR* dataset with domain knowledge and experience in the badminton sports. The first annotator assigned the labels, and then it was cross-validated by the second annotator.

B. Data Preprocessing

This study considers the raw signals acquired from the accelerometer, gyroscope, and magnetometer as input features collected from the body-worn IMU sensor network. The raw data is venerable to noise, such as motion artifacts. So, the raw data acquired were preprocessed using a median filter to eliminate the data's noise. Further, we normalized (min-max scaler) 48 features for the *BAR* dataset. The 48 features comprise three-axis of low noise & wide range accelerometer, gyroscope, and magnetometer sensors data. Due to the low measurement range of the low noise accelerometer sensor, most of the signals were clipped due to high acceleration (more than $\pm 2g$) from jerks and swift shots. Hence, such a phenomenon encouraged us to employ the wide/high range accelerometer raw signals, which enable us to capture those jerk and swift shots in nature. Next, we employed the sliding windowing technique for the raw features. It is vastly used in sensor-based human activity recognition problems among the signal processing techniques and extensively removes the motion or device artifacts from the signal dataset. We employed a sliding windowing with 50% overlap with a window size of 0.125 sec at a sampling rate of 512Hz and 25 Hz for the *BAR* and *DSA* datasets. We employed the majority voting for data labelling for each window segment to select the most activity labels that occurred within each window segment. The overlap-defined windows technique is better for extracting temporal patterns for micro-activities than the activity and event-defined windows [18]. Lastly, most badminton strokes are a composite of the jerk and swift actions.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(X_i^{actual\ value} - X_i^{predicted\ value})^2 \quad (4)$$

$$R^2\ \ Score = 1 - \sum_{i=1}^{N}\frac{(X_i^{actual\ value} - X_i^{predicted\ value})^2}{(X_i^{actual\ value} - mean(X_i^{actual\ value}))^2} \quad (5)$$

C. Evaluation Strategy and Setup

This section highlights the evaluation strategy and setup utilized to determine the overall performance of the *PerMTL* framework. We computed two evaluation metrics: ***mean squared error (MSE)*** and $\boldsymbol{R^2}$ ***score*** as shown in equations 4 and 5, respectively. Moreover, the *DBEL* module, we have re-designed each module by utilizing the exact hyperparameters used in the studies [14]. The motivation behind employing the same hyper-parameters is to maintain coupled experiment pipeline throughout the study. We employ a *60-20-20%* dataset split for training, validation and testing sets. The validation set was used to fine-tune the hyperparameters of the *MTL* pipeline, which is shown in table IV. The validation and test datasets were not utilized during the training phase.

VI. Results and Discussion

This section discusses and highlights the various analysis performed using the BAR dataset. Furthermore, the BAR dataset is collected in controlled and uncontrolled environment settings, as discussed above. Including a dataset collected in an uncontrolled environment introduces real-world heterogeneities such as stiff movements due to uneven playground, wind, lack of inattentiveness, etc., and contrasts the proposed framework's performance in the controlled environment setting. Furthermore, the dataset will help us to demonstrate the robustness of the *PerMTL* by overcoming the above challenges. Lastly, by employing the uncontrolled setting dataset, we would like to predict players' performance in the game

TABLE II: Badminton participants detail collected in a controlled setting (**LH** and **RH** represents left-hand and right-hand, respectively)

Participant	Gender	Expertise	Dominance Limb
PP-1	Male	Professional Player	LH
PP-2	Male	Professional Player	RH
IP	Male	Intermediate Player	RH
NP	Male	Novice Player	RH

TABLE III: Badminton participants detail collected in an uncontrolled setting (**RH** represents right-hand)

Participant	Gender	Dominance Limb
P-1	Male	RH
P-2	Female	RH
P-3	Male	RH
P-4	Female	RH
P-5	Female	RH
P-6	Male	RH
P-7	Male	RH

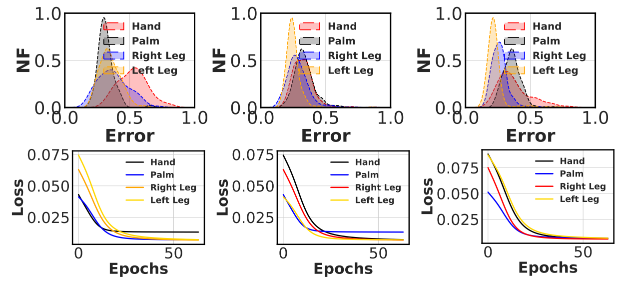
TABLE IV: Hyperparameters of MTL module

Hyper-parameters	Values
No. of maximum convolution layers	3
No. of filters in convolution layers	256, 196, 128
Convolution filter dimension	5x1,5x1,5x1
No. of maximum fully connected layers	2
No. of neurons in fully connected layers	32
Batch size	256
Max number of epochs	64

and demonstrate that the proposed framework can successfully predict the players' proficiency and cross-validate the players' performance based on the handcrafted scores reported in the [14]. Before proceeding, we employed convolution neural network (CNNs) layers as our shared layers to learn the feature representation of the raw data from each participant's limb. Due to the shift-invariant property, we employed CNNs layers to learn deep intricate and meaningful features from the shared feature space. We also listed the hyper-parameters employed for the MTL pipeline in Table IV. The following segments below exemplify and enumerate the scalability and generalizability characteristics of the *PerMTL* framework.

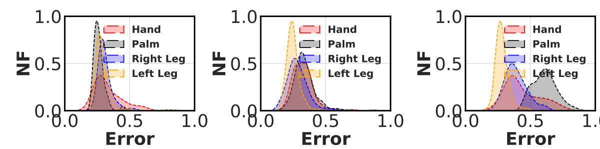
Moving forward, Fig. 3(a-c) shows the errors and losses obtained from each limb of the three participants (PP-2, IP and NP) collected in a controlled setting, which was examined w.r.t the PP-1. The loss graphs show that the network successfully learned each participant's limb's discrepancies and unique traits (style, speed, definitive and repetitive patterns, limbs movements, etc.). Furthermore, we noticed that the loss values reduce with each iteration of the experiment, which specifies that the network is trying to learn the feature representation of the limb data of the player. Finally, figures 5(a-g) shows the errors obtained from each limb of the seven players (P-1 - P-7) collected in an uncontrolled setting, which was examined w.r.t the PP-2. We plotted the probability density function (*pdf*) plots to exhibit the error values obtained. Moreover, the y-axis corresponds to the normalized frequency (probability of error occurrence) on the scale of 0-1. The x-axis corresponds to the error scores; the higher the error scores, the *pdf* plot will be closer to 1 and vice versa. Moreover, the higher overlap between *pdf* plots denotes higher synchronous occurring of errors between the limbs.

To exemplify the proposed framework's robustness and adaptability, we introduce ***cross-person validation*** strategy, where we consider another professional player, PP-2 as the reference player similarly shown in the state-of-the-art preliminary [14] work. We performed the experiments with the players collected in an uncontrolled environment setting w.r.t the PP-2 shown in Figs. 4(a-c). The motivation is to showcase that the *PerMTL* successfully learned the distinctive traits irrespective of the dominant limbs where we experimented with both left-handed and right-handed dominant players. We determined that P-3 & P-6 and P-2 & P-4 are the best players among other male and female players, respectively. We noticed



(a) NP (b) IP (c) PP-2

Fig. 3: (a-c): Error and loss plots for all the players w.r.t. the PP-1, where NF = Normalized Frequency



(a) NP (b) IP (c) PP-1

Fig. 4: (a-c): Error plots of the players w.r.t. the PP-2

that the probability of an error occurring *pdf plots* is low compared to the other players. To bolster our propositions, the error plots shown in Figs. 5 and *MSE* and R^2 scores shown in Table VII, endorse and contemplate the similar trends. Moreover, we can also determine that for the P-2, P-3, P-4 and P-6, the error *pdf* plot has higher overlapping (probability of error occurrence), which determines that the limbs' movements were similar synchronous. It is because, for good players, the movements of the limbs are coherent and highly coordinated while playing badminton shots. Furthermore, we used the recorded videos of each data collection session as the ground truth and manually validated our conclusions and findings. Lastly, we are confident that the ***PerMTL*** framework can learn the micro-activities discrepancies and errors among the limbs and can be used to determine the players' proficiency irrespective of the real-world heterogeneities and challenges.

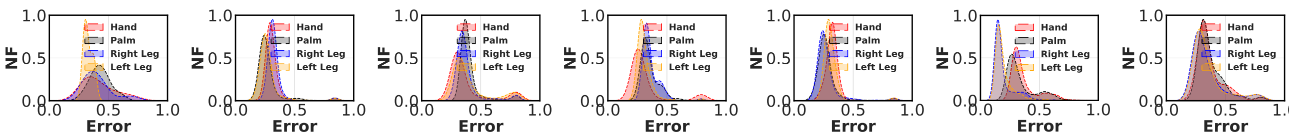
Next, we examined an intrigue phenomena, Figures 3(c) and 4(c), where each experiment examined w.r.t the PP-1 and PP-2, respectively. The study shows that the PP-1 and PP-2 have similar trends in both experiments. To strengthen our proposition, Figs. 3, PP-2 has the highest vulnerability probability of receiving a low error score for the left leg,

TABLE V: Mean Squared Error and R-Squared Score of the players w.r.t the PP-2

Players	Sensor Position	MSE	R^2 Score
PP-1	Hand	0.0057	0.5580
	Palm	0.0064	0.6266
	Right Leg	0.0038	0.6952
	Left Leg	0.0033	0.4786
IP	Hand	0.0011	0.7570
	Palm	0.0021	0.7131
	Right Leg	0.0035	0.8189
	Left Leg	0.0025	0.7609
NP	Hand	0.0030	0.4709
	Palm	0.0042	0.4358
	Right Leg	0.0034	0.5265
	Left Leg	0.0027	0.4598

TABLE VI: Mean Squared Error and R-Squared Score of the players w.r.t the PP-1

Players	Sensor Position	MSE	R^2 Score
PP-2	Hand	0.0038	0.5979
	Palm	0.0020	0.5343
	Right Leg	0.0054	0.5775
	Left Leg	0.0050	0.4924
IP	Hand	0.0049	0.5582
	Palm	0.0032	0.4924
	Right Leg	0.0038	0.4378
	Left Leg	0.0041	0.2617
NP	Hand	0.0069	0.3381
	Palm	0.0058	0.2938
	Right Leg	0.0054	0.3689
	Left Leg	0.0050	0.3839



(a) P-1 (b) P-2 (c) P-3 (d) P-4 (e) P-5 (f) P-6 (g) P-7

Fig. 5: (a-g): Error plots all the players w.r.t. the PP-2, where NF = Normalized Frequency

whereas palm has the low vulnerability probability of receiving a high error score. Compared to, in Figure 4(c), PP-1 has the highest vulnerability probability of receiving a low error score for the left leg, whereas palm has the low vulnerability probability of receiving a high error score. To bolster our discussion, we computed MSE and R^2 $scores$ shown in Tables V and VI, and undoubtedly we examined similar trends on the *MSE* and R^2 *scores*. We believe that such phenomena are because the PP-1 & PP-2 are right and left-handed players. The style, swing, posture and muscle exertion are changed w.r.t the players' dominant limbs. We are assertive that *PerMTL* framework can successfully learn the minute and distinctive traits of the players irrespective of the dominant limb, and also strengthens the generalizability and adaptability characteristics of the *PerMTL* framework.

We benchmark the *PerMTL* and highlight the hardware (time) resource consumption shown in Table VIII. The motivation is to identify and showcase any performance bottlenecks due to any PyTorch operations. Another motivation is to mitigate the high computational resources challenge faced in the *DBEL* module. In Table I, *B(total)*, shows the total time consumed for an experiment. Comparatively, in Table VIII, *A (total)*, shows the total time consumed from ***CPU and Cuda (GPU) processors*** took ***99.204 ms*** and ***88.825 ms***, respectively. Furthermore, the execution time of the overall *MTL* pipeline is ***3.24 mins*** with total trainable parameters ***511720***, which is comparatively very less than the *DBEL* module. We are assertive that the inference time drastically reduced the computational complexity and encourages us to deploy the inference module on edge devices with limited memory usage for real-time inference.

A. Comparison with Baseline

To better demonstrate the efficacy of our proposed approach, we examine a public dataset to showcase how well the proposed algorithm scales to different scenario heterogeneities.

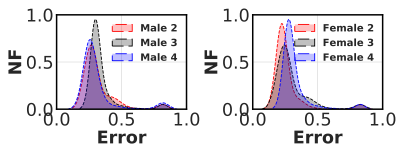
TABLE VII: Mean Squared Error and R-Squared Score for each player w.r.t the PP-2

Players	Sensor Position	MSE	R^2 Score
P-1	Hand	0.0296	0.4316
	Palm	0.0321	0.3887
	Right Leg	0.1021	0.3114
	Left Leg	0.0934	0.3499
P-2	Hand	0.0181	0.7120
	Palm	0.0099	0.7286
	Right Leg	0.0128	0.7341
	Left Leg	0.0198	0.6896
P-3	Hand	0.0242	0.7545
	Palm	0.0368	0.7722
	Right Leg	0.0612	0.8056
	Left Leg	0.0199	0.7557
P-4	Hand	0.0271	0.6792
	Palm	0.0106	0.6533
	Right Leg	0.0335	0.6890
	Left Leg	0.0183	0.7153
P-5	Hand	0.0364	0.5270
	Palm	0.0172	0.5344
	Right Leg	0.0216	0.4856
	Left Leg	0.0191	0.4567
P-6	Hand	0.0081	0.7120
	Palm	0.0095	0.7286
	Right Leg	0.0028	0.7341
	Left Leg	0.0198	0.6976
P-7	Hand	0.0242	0.5545
	Palm	0.0368	0.5722
	Right Leg	0.0612	0.6056
	Left Leg	0.0199	0.5557

TABLE VIII: *Layer-wise time complexity* profiling of *PerMTL*

Layers	CPU Time	Cuda Time
CNN1D + Maxpool1D + + BatchNormal1D - (i)	$\approx 1.2842\ ms$	$\approx 13.982\ ms$
CNN1D + Maxpool1D - (ii)	$\approx 1.1022\ ms$	$\approx 10.987\ ms$
CNN1D + Maxpool1D - (iii)	$\approx 1.112\ ms$	$\approx 10.980\ ms$
A (Total)	$\approx 99.404\ ms$	$\approx 88.825\ ms$

1) ***Publicly available dataset:*** We chose the Daily, and Sports Activity dataset [19] (DSA) has a tri-axial accelerometer, magnetometer and gyroscope sensors data from eight users (four males and females). The sensors were placed on five body positions and collected at 25 Hz. They have performed



(a) Male Error (b) Female Error

Fig. 6: (a-b): Error plots for all the users w.r.t. the Male 1 and Female 1, respectively for DSA dataset

19 different activities- jumping, standing, sitting, lying on the back and right, playing basketball, etc. For the experiment, we assumed the first player from users list [19] (female and male) as our reference player. Hence, our proposition for the DSA dataset is based on the result we obtained through our experiments. Moreover, we perform a user-level error propagation to showcase how well *PerMTL* performs to evaluate and determine the users' performance for activities of daily living (ADLs) and instrumental activities of daily living (IADLs). We implemented the same experiments as the BAR dataset and reported the findings below. Interestingly, we observed that each male and female user showcased similar traits. We obtained R^2 Score of ***80.28%*** and ***79.12%*** for female and male users, respectively. Similar trends can be observed in the error figures that the overlap probability between the error plots is high, shown in figures 6 (a-b). Furthermore, we obtain ***96.89%*** and ***97.34%*** F1-score for male and female players respectively by employing the classification module reported in [14]. Lastly, from the error plots and F1-score results, we are assertive that the scale of expertise in performing ADLs and IADLs is mostly proximate and identical among the male and female players and also the error plots shows high overlapping probability among the players.

VII. Conclusion

This paper proposes an end-to-end BSN-driven framework, ***PerMTL*** that enables us to capture the distinctive traits between players' limbs for collective data inference and achieve R^2 Score of $\approx$ ***82%*** in predicting the players' performance. We performed a comprehensive analysis of our multi-task learning paradigm, and we have showcased that *MTL* can generate better error representation for the limbs. We also demonstrated that the proposed framework precisely learns the shared feature representation for each player's limb and will be befitting to estimate the error between the player's limbs w.r.t the professional players. We believe that the model's error can be used as feedback for the player to improve the performance in the game as it will help the users determine the most error-prone limb of the player while playing the game. Moreover, we explored and showcased the adaptability, robustness and scalability characteristics of the *PerMTL* framework. We showcased that the *PerMTL* can be employed across the different domains and can be scaled to ***user-level error estimation***. Lastly, we benchmark our proposed framework to showcase a substantial overall reduction in time computational cost. In the future, we would like to deploy the inference node in a resource constraint environment. Lastly, we would like to examine individual characteristics such as height, weight, speed, tiredness, etc., affecting the players' overall performance.

VIII. Acknowledgement

This research is supported by the NSF CAREER Award #1750936, REU Site grant $\#CNS - 2050999$ for Smart Computing and Communications and U.S. Army Grant $\#W911NF2120076$.

References

[1] S. Ishimaru, K. Hoshika, K. Kunze, K. Kise, and A. Dengel, "Towards reading trackers in the wild: detecting reading activities by eog glasses and deep neural networks," in *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*, 2017, pp. 704–711.

[2] N. Y. Hammerla, S. Halloran, and T. Plötz, "Deep, convolutional, and recurrent models for human activity recognition using wearables," *arXiv preprint arXiv:1604.08880*, 2016.

[3] H. S. Hossain, M. A. A. H. Khan, and N. Roy, "Soccermate: A personal soccer attribute profiler using wearables," in *2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. IEEE, 2017, pp. 164–169.

[4] Spektacompowerbat, https://spektacom.com/product/, 2020.

[5] H. Zhao, S. Wang, G. Zhou, and W. Jung, "Tenniseye: tennis ball speed estimation using a racket-mounted motion sensor," in *Proceedings of the 18th International Conference on Information Processing in Sensor Networks*, 2019, pp. 241–252.

[6] T. Steels, B. Van Herbruggen, J. Fontaine, T. De Pessemier, D. Plets, and E. De Poorter, "Badminton activity recognition using accelerometer data," *Sensors*, vol. 20, no. 17, p. 4685, 2020.

[7] M. Salim, H. Lim, M. Salim, and M. Baharuddin, "Motion analysis of arm movement during badminton smash," in *2010 IEEE EMBS Conference on Biomedical Engineering and Sciences (IECBES)*. IEEE, 2010, pp. 111–114.

[8] T. Jaitner and W. Gawin, "Analysis of badminton smash with a mobile measure device based on accelerometry," in *ISBS-Conference Proceedings Archive*, 2007.

[9] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[10] O. Barut, L. Zhou, and Y. Luo, "Multi-task lstm model for human activity recognition and intensity estimation using wearable sensor data," *IEEE Internet of Things Journal*, 2020.

[11] C. Zalluhoglu and N. Ikizler-Cinbis, "Collective sports: A multi-task dataset for collective activity recognition," *Image and Vision Computing*, vol. 94, p. 103870, 2020.

[12] M. Seiffert, F. Holstein, R. Schlosser, and J. Schiller, "Next generation cooperative wearables: Generalized activity assessment computed fully distributed within a wireless body area network," *IEEE Access*, vol. 5, pp. 16 793–16 807, 2017.

[13] A. Khan, J. Nicholson, and T. Plötz, "Activity recognition for quality assessment of batting shots in cricket using a hierarchical representation," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 3, pp. 1–31, 2017.

[14] I. Ghosh, S. R. Ramamurthy, A. Chakma, and N. Roy, "Decoach: Deep learning-based coaching for badminton player assessment," *Pervasive and Mobile Computing*, p. 101608, 2022.

[15] I. Ghosh, S. R. Ramamurthy, and N. Roy, "Stancescorer: A data driven approach to score badminton player," in *2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. IEEE, 2020, pp. 1–6.

[16] A. Argyriou, T. Evgeniou, and M. Pontil, "Multi-task feature learning," *Advances in neural information processing systems*, vol. 19, 2006.

[17] Y. Zhang and Q. Yang, "A survey on multi-task learning," *arXiv preprint arXiv:1707.08114*, 2017.

[18] Y. Guan and T. Plötz, "Ensembles of deep lstm learners for activity recognition using wearables," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 2, pp. 1–28, 2017.

[19] K. Altun and B. Barshan, "Human activity recognition using inertial/magnetic sensor units," in *International workshop on human behavior understanding*. Springer, 2010, pp. 38–51.