

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2021.DOI

On-Device Lumbar-Pelvic Movement Detection Using Dual-IMU: A DNN-based Approach

YUXIN ZHANG¹, (Member, IEEE), PARI DELIR HAGHIGHI¹, FRADA BURSTEIN¹, LINA YAO², (Member, IEEE), AND FLAVIA CICUTTINI³

¹Faculty of Information Technology, Monash University, Caulfield, VIC 3145 Australia

²School of Computer Science and Engineering, UNSW Sydney, Sydney, NSW 2052 Australia

³School of Public health and Preventive Medicine, Monash University, Melbourne, VIC 3004 Australia

Corresponding author: Yuxin Zhang (e-mail: yuxin.zhang@monash.edu).

This work is partially sponsored by Monash Institute of Medical Engineering.

ABSTRACT Lumbar-pelvic movements (LPMs) are generally performed in the clinical setting to identify limitations in a range of movements. Continuous monitoring of these movements can provide real-time feedback to both patients and medical experts with the potential of identifying activities that may precipitate symptoms of low back pain (LBP) as well as improving therapy by providing a personalised approach. Recent advances in mobile computing technology and wearable sensors have paved the way for developing mobile physical activity monitoring applications with more advanced and complex algorithms, such as deep neural network (DNN) based models. However, there is a lack of prior studies that focus on real-time LPMs detection with multimodal sensory data. Meanwhile, most research in the area of body movement detection do not consider the potential transition logic of the constituent low-level body movements (e.g., LPMs) within their corresponding high-level physical activity. This information could significantly increase accuracy of detection results. Further, current studies mainly perform deep learning-based movement detection in the cloud (or a backend server) that could increase network bandwidth and response time. To address these limitations, this paper proposes a novel LPMs detection approach using an enhanced and adapted hybrid DNN model, which includes a convolutional neural network followed by a long-short term memory recurrent network (CNN-LSTM), and performs detection locally on the mobile device. The results of a comparative evaluation of the proposed model and baseline models are described. We also introduce a set of domain-specific pre-defined rules, based on the transition logic information, to reconstruct the detection outputs to further improve the detection accuracy.

INDEX TERMS Deep Neural Networks, Domain Adaptation, Lumbar-Pelvic Movement Detection, Mobile Platform, On-device processing, Temporal-wise Attention

I. INTRODUCTION

LUMBAR-PELVIC movements (LPMs) play a key role in the development of low back pain (LBP) and are important targets in the management of LBP [1]. The LPMs include flexion, extension, left and right lateral flexion, right rotation and left rotation. Dissecting how an individual undertakes activities of daily living such as cleaning or gardening, or recreational activities that may precipitate LBP in that person can provide granular and useful physiological parameters (such as speed, acceleration, intensity and range of movement) about the LPMs and the corresponding activity. Quantifying these parameters provides the opportunity

to identify movements most likely to precipitate LBP and also are targets for therapy, providing the opportunity for a personalised approach to the management of LBP. This is not currently possible in clinical practice.

Out-of-hospital physical activity monitoring is one of the most widely discussed research topic among all the existing telehealth applications [2]. Recent advances in mobile technology and wearables have made objectively continuous monitoring of LPMs in out-of-hospital settings possible. This has the potential to empower patients through provision of objective data and the ability to co-develop personalised self-management goals in collaboration with their healthcare

provider. Movement detection systems of LPMs can also be further extended to allow remote monitoring by healthcare providers and clinicians. Exercises can also be adapted based on real-time feedback from the patients. Patients may participate in the development of their programs through shared decision making and undertake activities at convenient locations reducing travel costs and waiting time in clinics.

LPMs are low-level body movements. These low-level movements form high-level daily physical activities, such as cleaning, cooking, jogging and cycling [3]. One of the major differences between high-level physical activities and low-level body movements is the level of granularity. High-level physical activities can be performed for a relatively longer time and consist of a series of short time and sequential low-level body movements.

Different individuals may perform the same high-level physical activity in different ways. Thus, this research developed an approach that focuses on detecting the low-level body movements (LPMs in this case), to provide more granular details about an activity (e.g., the number of times an individual performing right rotation movements while standing). By further using data fusion techniques [4], [5], the physiological parameters of specific low-level body movements, such as speed and acceleration, can be also retrieved based on the detection results. In this way, the high-level physical activities can be quantified and described by the physiological parameters of their constituent low-level body movements (i.e., understanding how the corresponding high-level physical activity is performed).

Moreover, existing approaches also failed to consider the logical sequence (i.e., the transition logic) of low-level body movements within a high-level activity. For example, in a high-level physical activity like playing basketball, a player has to pick up the ball and raise it above the shoulder before he/she can throw it to the basket. As this example shows, the body movements are sequential and have a higher transition frequency compared to the high-level physical activities. Additionally, the relationship between the adjacent body movements within a high-level physical activity is strong and important. There is normally some prior knowledge about the order of the adjacent body movements in certain high-level physical activities (e.g., first picking up the ball and then throwing it in playing basketball). Knowledge about the body movement transition logic is not generic, and commonly acquired based on the experience of domain experts and literature. It varies from domain to domain and changes from case to case. Developing a method for detecting and monitoring low-level body movements as parts of high-level activities is therefore a useful research challenge, which can inform a better and more personalised clinical practice.

Thus, this work proposed a domain specific rule-based method to find and model the relationship between adjacently detected body movements (i.e., the logic/order of body movements transition within their constituent high-level physical activity), by consulting the medical experts and reviewing literature. This method can potentially correct the wrongly

detected LPMs to improve the detection accuracy.

Another limitation of the exist physical activity recognition methods is that while they use mobile devices to collect sensory data, they perform inference and detection at the backend server or in the cloud, mainly when using deep learning-based approaches [6]. This type of computing infrastructure increases response time in real-time systems due to the need for a massive data transmission between sensors, the mobile device and the server [7].

Taking advantage of edge computing to perform deep learning and detection locally on-device (on the smartphone) can significantly reduce the data transmission and response time in cloud computing. It can also address the data privacy problem by locally processing patients' sensitive medical data. Therefore, edge computing is a promising computing paradigm for out-of-hospital body movement monitoring applications.

Despite its importance and urgency, there is a scarcity of reliable deep learning-based approaches for detecting LPMs on mobile devices in an out-of-hospital setting using wearable sensors and most existing deep learning-based systems have limited ability to run movement detection on mobile devices and support edge computing. To address these research gaps, we propose an enhanced and adapted convolutional neural network integrated with a long-short term memory recurrent network (CNN-LSTM) model which is also using domain adaptation technique and global temporal-wise attention mechanism to detect six key LPMs on mobile device in real-time, by using dual inertial measurement units (IMUs) sensory data. The paper also provides a detailed comparative evaluation of our proposed CNN-LSTM model with benchmark models on both PC and mobile platforms for LPMs detection. As part of this detection approach, we also introduce a rule-based method for reconstruction of movement segments to improve the on-device detection accuracy of the proposed model. The contributions of this research are as follows:

- 1) An enhanced and adapted CNN-LSTM model, based on domain adaptation technique and global temporal-wise attention mechanism, for improving dual-IMUs sensory data inference performance (i.e., accuracy, processing speed and power consumption).
- 2) Combining the proposed model with a real-time on-device rule-based method to reconstruct movement segments and reclassify misclassified segments to improve the overall on-device inference accuracy.

The remainder of this paper is organised as follows. We introduce the related works in Section II. In Section III, we thoroughly explain the proposed framework, the enhanced CNN-LSTM model with domain adaptation technique and global temporal-wise attention mechanism and the concept of the post-detection rule-based reconstruction method. We illustrate the experimental evaluation experiments' design in Section IV and discuss the comparison of performances in Section V. Finally, we present the conclusions and elaborate

on the limitations of this research as well as the future works in Section VI.

II. RELATED WORKS

Various wireless sensors and wearables have been used for measuring LPMs. One of the most widely used sensors is IMUs [8]. The IMUs are normally composed of a 3-axis accelerometer and a 3-axis gyroscope, and sometimes a 3-axis magnetometer. IMUs have been proven to be reliable and accurate for many body movement measurement tasks [8], [9]. Based on the number and type of the IMUs, these sensing systems can be further divided into three categories: single IMU based systems, dual-IMU based systems and multi-IMU based systems.

Single IMU based systems are mostly designed for spinal posture and simple spinal movement monitoring. Most of these systems have bio-feedback functions to help individual to improve their awareness of inappropriate spinal postures and movements, such as Spineangel [10], Lumo Lift [11], and Upright posture trainer [12].

Dual-IMU based systems enhance the possibility of measuring complex body movements with better accuracy. This is because single IMU system can only reflect a single point movement in three dimensional spaces. However, the lumbar-pelvic area is a soft surface which moves simultaneously with the spine [13]. Thus, the dual-IMU based systems provide an additional measurement point within the lumbar-pelvic area, which can better simulate the LPMs and provide more accurate measurements. Several dual IMU sensor-based systems have been developed for monitoring LPMs, including ViMove [4], Valedo Motion [14], and RIABLO [15].

Multi-IMU based systems usually have more than three IMU sensor nodes attached on different parts of human body to measure different body movements [16], [17]. Since many of these systems sew sensor nodes on the garments, this type of system is also known as the garment integrated sensing system, such as the Xsens MVN system [18]. Comparing to the dual-IMU based systems, they have more three-dimensional measurement points attached on the human body parts. Thus, the multi-IMU based systems are capable of providing more accurate measurement for various complex body movements [19].

This research aims to detect the six standard LPMs in real-time using dual-IMU sensors. It can be seen that single IMU based system is not capable of providing accurate measurement for these LPMs. On the other hand, although the multi-IMU based system is capable of providing more accurate measurement for the LPMs, the calibration steps for this type of system can be complicated [20], and more sensors may not necessarily increase the measurement accuracy. Moreover, attaching more sensors on human body may make the users feel uncomfortable for long-time monitoring [13]. Therefore, this study chooses ViMove [4], a dual-IMUs based sensor system available on the market, for LPM data collection. It is capable of measuring all the six standard LPMs with high accuracy. ViMove also has been clinically validated against

the VICON system [21], which is the gold standard in related research fields.

Recently, the HAR research community has started to apply DNN for more complex modelling and inference tasks. The two most widely used DNN models are CNN (convolutional neural network) and RNN (recurrent neural network) [22]. CNNs are capable of capturing the local connections of multimodal sensory data [23]. Researchers normally use CNNs to extract the multimodal features. The sensing modality fusion can be categorised into three types: early fusion (EF), sensor-based late fusion (SB-LF), Channel-based late fusion (CB-LF) and Shared filters hybrid fusion (SF-HF) [24].

On the other hand, RNNs are suitable for extracting temporal dependencies and learning information incrementally through time intervals, so they are normally used for analysing streaming data (i.e., time-series data). Among all the existing RNNs, LSTM (Long Short Term Memory) network has gained increasing attention because it enables the gradients to easily flow through the time, which solves the gradient vanishing/exploding problems of the tradition RNNs [25]. There is also a research trend that combines the advantages of both CNNs and RNNs together into a hybrid model to explore different views of temporal dynamics [23].

Many studies have taken advantage of these DNN models to recognise various high-level physical activities with higher accuracy, including 1D CNNs [26], CNN-LSTM [27] and Bidirectional LSTM [28]. However, existing studies present three major drawbacks: 1) unlike the CNN models, LSTM-based DNN models are computationally intensive [25]. This is why most of previous studies on resource-constraints systems, such as embedded device and smartphones, are only using CNNs. Recent advances in smartphone hardware and DNNs optimisation technologies on mobile platforms have significantly overcome this obstacle [29]. With increasing computing power on mobile devices, computationally intensive models such as LSTM can be run on a mobile device like a mobile phone. 2) most of these DNN models only focus on the sensor-level (e.g., accelerometer, gyroscope and magnetometer in the IMU device) multimodal data fusion [24]. There is a lack of studies considering device-level feature augmentation with DNNs. 3) In specific domain, two adjacent body movements may follow certain transition logics. However, existing studies fail to consider these potential transition logics between the two adjacent body movements.

III. THE PROPOSED APPROACH

This study proposes a DNN based mobile approach for detecting LPMs by using dual-IMUs. The paper describes the design and development of this approach and its performance evaluation. The proposed approach overview is shown in Figure 1. It consists of three main procedures, including Data Pre-Processing, LPM Detection and Rule-based Reconstruction. This section explains the functionality and components of each procedure.

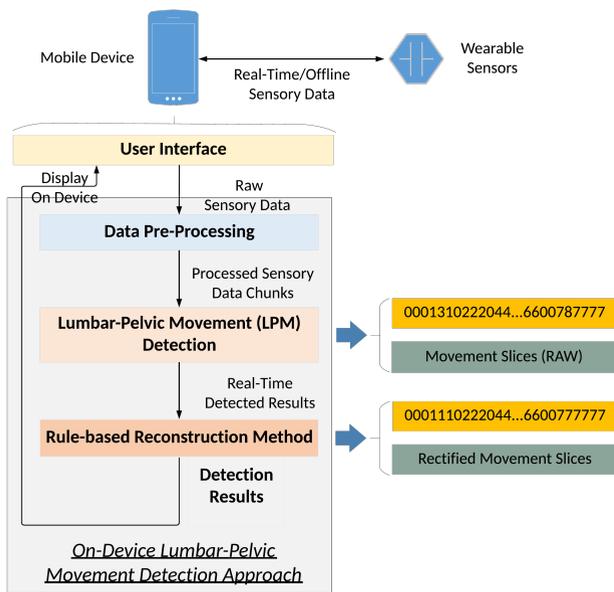


FIGURE 1. Approach Overview

A. DATA PRE-PROCESSING

Data pre-processing is responsible for transforming raw multivariate time series data into DNN model processable time series data. The standard data pre-processing procedures involve de-noising, calibration, unit conversion, normalisation and synchronisation [30]. Nowadays, some sensors run these pre-processing procedures on-board [4] and directly present processed data to the users, while other sensors only provide raw data. Since mobile devices are resource-constraint, only necessary data pre-processing procedures should be selected and run on-device. The on-device processing speed and accuracy are two key values for selecting necessary procedures. Another important task for the data pre-processing is to slice the data into data chunks for further detection and classification. The slicing procedures are controlled by slicing window size and overlap rate. Detailed pre-processing procedures used in this paper are illustrated in Section IV.

B. LPM DETECTION MODEL

LPM detection is responsible for the classification of pre-processed data into different LPM categories. Generally, this procedure consists of feature extraction on pre-processed data and running a pre-trained classification model on extracted features [31]. This paper uses DNN based models to perform feature extraction and classification. These DNN-based models usually contain a number of hidden layers that can be used for feature extraction and a fully connected layer as well as an output layer that are used for classification. Previous results show that DNN-based models outperform models that are fed with hand-crafted time-domain and frequency-domain features [32]. Therefore, our work directly uses the pre-processed data and transforms it into the input shape of the DNN models for the detection of LPMs. The

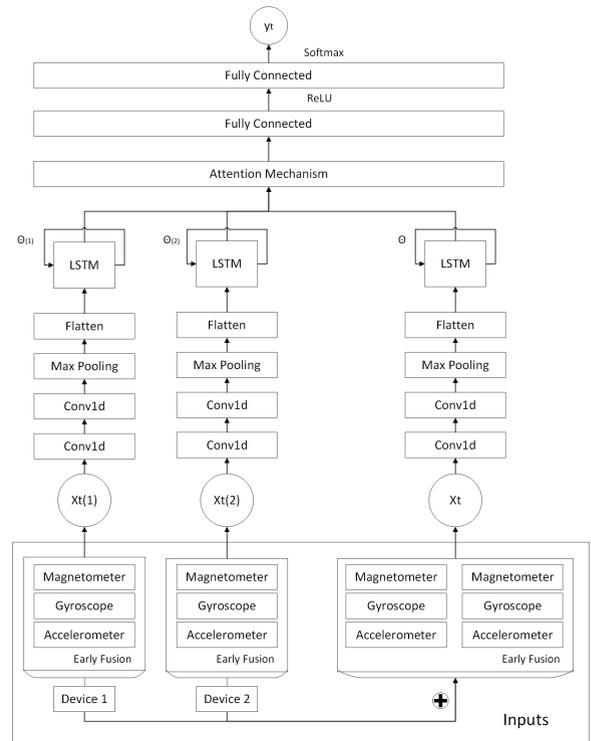


FIGURE 2. Proposed CNN-LSTM with Domain Adaptation and Global Temporal-Wise Attention

implementation of the DNN model should consider the actual requirement and specification of the detection system.

In this paper, we propose an enhanced and adapted CNN-LSTM model for on-device LPM detection. This model has two main components including: domain adaptation with early fusion and CNN-LSTM with global temporal-wise attention mechanism. The detailed architecture of the proposed model is shown in Figure 2. We choose this architecture based on the following reasons. Firstly, LSTM has a better performance in terms of dealing with time series data. Secondly, using CNN to generate an appropriate compressed representation of the dataset before fed into the LSTM network, which can significantly improve the on-device inference speed comparing to other LSTM-based models (A detailed comparison of the performance is shown in Section V). Lastly, the layers are empirically optimised for this specific task (i.e., more layers do not improve the accuracy any further and slow down the real-time processing speed).

1) DOMAIN ADAPTATION WITH EARLY FUSION

This study uses dual-IMUs based ViMove PC software to collect the data. Each IMU has one accelerometer, one gyroscope and one magnetometer. Altogether, there are 18 features including 6 readings of 3-axis from two accelerometers, 6 readings of 3-axis from two gyroscopes and 6 readings of 3-axis from two magnetometers. Due to the fact that the two IMUs are identical but the features of these two IMUs' data are different for the same LPM, we consider each IMU

as a domain and apply a domain adaptation technique to extract domain invariant super features that could benefit the classification in individual domains. This domain adaptation technique was proposed in [33]. It partitions the model to deal with common patterns (exists in all domains) and domain specific patterns. Assuming there are N domains (i.e., analogous to the two IMUs in our case), the idea of this method is to expand the feature space with a duplicated $(N + 1)$ th domain-combined input $< domain1 + \dots + domainN + domain(N + 1) >$ that has the general features across all N domains. By using this $(N + 1)$ th network to train this new input, the overall model not only can learn domain-specific features but also the domain-invariant super features across all domains. This method has been successfully applied in many NLP tasks. With the similar idea, we adopt this domain adaptation technique from NLP to LPMs detection tasks and proposed a model that is able to extract super features of the three individual sensors within each IMU device (i.e., domain). Our model considers IMU device 1 and IMU device 2 as two domains. It consists of $N+1$ CNN-LSTMs: two CNN-LSTMs are respectively trained only on the corresponding domains (the two IMU devices), and an additional CNN-LSTM is trained on all domains (device1 and device2).

On the other hand, Noori *et al.* [34] proved that an early fusion method outperforms a sensor-based late fusion method for CNN-based models in sensory data inference (Early fusion refers to data fusion at input level in the first convolutional layer, while late fusion refers to data fusion in a fully connected layer after feature extraction). Therefore, we decide to use early fusion to fuse the three individual sensory data (including accelerometer, gyroscope and magnetometer) within one device and the domain adaptation technique for feature augmentation before the training process.

2) CNN-LSTM WITH GLOBAL TEMPORAL-WISE ATTENTION MECHANISM

The main architecture of the proposed model is a CNN-LSTM neural network. The processed inputs are fed into two stacked time-distributed 1-D convolutional layers followed by 1-D max pooling and dropout layers (drop probability is set to be 0.5 to avoid overfitting). All the 1-D convolutional layers use Rectified Linear-Units (ReLU) as activation functions. The temporal max pooling operation is used to keep salient information in the outputs of convolutional layers, while reduce the dimensions. The output of the CNN is then flattened and passed into LSTM network to capture time dependencies on features. A drop probability of 0.5 is also applied on this layer to avoid overfitting. The domain-specific features (device-wise features) and all-domain features (common features) that are captured by the CNN-LSTM are concatenated and fed into a process called global temporal-wise attention mechanism to obtain attention map.

Attention mechanism has been widely used on neural machine translation (NMT) tasks for capturing the context from all possible feature combinations [35]. Due to similarity between NMT tasks and HAR tasks, the utilisation of

attention mechanism for HAR has been explored in [36]. The uses of attention mechanism for HAR are to 1) make the model more interpretable [37], which is normally used in wireless body sensor network systems to identify which sensors are active during specific tasks such as running, and 2) incentivise the model to generate and place the weights on the context which is relevant for classification decision to improve the performance of the model [36]. In our case, only two IMUs are used for detecting LPMs. Both of the IMUs are active when individuals performing LPMs. Therefore, the focus of this paper is to utilise the attention mechanism to improve the overall model performance.

To do this, we combine the LSTM networks with the global temporal-wise attention mechanism. This mechanism is used to create context vectors using the combination of past and current hidden state outputs of the LSTM layer and learn parameters to rank them based on their importance for classifying the corresponding LPMs in the sliding window.

$$a_{ts} = \frac{\exp((h_t)^T \cdot h_s)}{\sum_s \exp(h_t \cdot h_s)} \quad (1)$$

$$c_t = \sum_s a_{ts} \cdot h_s \quad (2)$$

$$h_t = \tanh(W[c_t; h_t]) \quad (3)$$

Equation (1) above is used to obtain the attention weight a_{ts} . h_t refers to the current hidden state of the precedent LSTM layer, h_s refers to the past context and c_t is the context vector. (2) represents a weighted summation based on the relative importance of past context. The ranking (attention vector) is calculated in (3) above. It creates a representation of the past context and current in the sliding windows, which are used as feature inputs of the two stacked dense layers (i.e., fully connected layers) for classification. W represents the 2D matrix in the linear layer.

C. DOMAIN SPECIFIC RECONSTRUCTION METHOD

The rule-based reconstruction method is developed to reduce detection errors and re-detect mis-detected movements based on pre-defined rules [38].

In sequential LPMs detection, the inputs are sliced into overlapped data chunks, so the outputs of detection are dependent: previous detection result has relationships with the subsequent detection results. Based on the relationships, three pre-defined rules are generated. The rule-based reconstruction method uses the pre-defined rules to rectify the wrongly detected results, which makes them suitable for further analysis.

Rule 1 is designed to eliminate the error that a single wrong classification exists in a correct classification series. This is a generic rule which can be used for any other movement detection scenarios. Rule 2 is designed to determine the category of transition inputs correctly. This rule can be applied to other sequential movement detection tasks. Rule 3 is designed specifically for sequential LPM movements with a

static position (e.g., standing straight) in between, which can be found in rehabilitation progress or home-based exercises.

Rule 1: If the previous classification (O_{t-1}) is the same as the subsequent classification (O_{t+1}), current classification (O_t) should be the same as these two classifications. This is because each input data is overlapped data with an overlap rate of $a_{overlap}$. This rate is a pre-set value determined by the requirement of different tasks. The previous input (I_{t-1}) and the subsequent input (I_{t+1}) both contain parts of the current input (I_t). Then, the current input can be represented as Equation (4) and (5). Here, the $a_{overlap} \cdot I_{t-1}$ means the latter $a_{overlap}$ part of I_{t-1} and $a_{overlap} \cdot I_{t+1}$ means the former $a_{overlap}$ part of I_{t+1} . Both of them partially contain information of I_t . Most related studies set the overlap rates greater than 50% [28], [39]. When $a_{overlap}$ is greater than 50%, the residual C is equal to or greater than zero. It means current input can be fully represented by using previous input and subsequent input. Therefore, the current classification is very unlikely to be different when the previous classification is equal to the subsequent classification. It should be the same as the previous classification and the subsequent classification.

$$I_t = a_{overlap} \cdot I_{t-1} + a_{overlap} \cdot I_{t+1} - C \quad (4)$$

$$C = \begin{cases} (2 \cdot a_{overlap} - 1) \cdot I_t, & \text{when } a_{overlap} < 50\% \\ 0, & \text{when } a_{overlap} = 50\% \\ I_{t-1} \cap I_{t+1}, & \text{when } a_{overlap} > 50\% \end{cases} \quad (5)$$

Rule 2: If the current classification equals either previous classification or subsequent classification, it is very likely that the current status is in movement transition (e.g., changing from flexion to extension). In order to correctly classify the current status into the right category, the current input is separated into two new inputs, $I_{t_previous}$ and $I_{t_subsequent}$, by up-sampling based on the overlap rate $a_{overlap}$. Then these two new inputs are sent back to the LPM detection and classification module to generate two new classifications, $O_{t_previous}$ and $O_{t_subsequent}$. When $O_{t_previous}$ equals O_{t-1} and $O_{t_subsequent}$ does not equal O_{t+1} , it can be assumed that the current classification has a stronger relationship with the previous classification, so the current classification is set as the value of $O_{t_previous}$, and vice versa. For any remaining situation, current classification stays the same (i.e., equals to O_t).

Rule 3: In addition to Rule 2, we consider if a movement transition is detected (i.e., $O_t \neq O_{t-1}$), then O_{t-1} and O_{t+1} are compared. If both O_{t-1} and O_{t+1} are not equal to zero, (zero means the category of static status), Rule 2 is performed with the following new condition: If reclassified previous output $O_{t_previous}$ equals zero and reclassified subsequent output $O_{t_subsequent}$ equals O_{t+1} , then O_{t-1} is corrected as zero (i.e., static status) and vice versa. For any remaining situations, the current classification stays the same (i.e., equals to O_t).

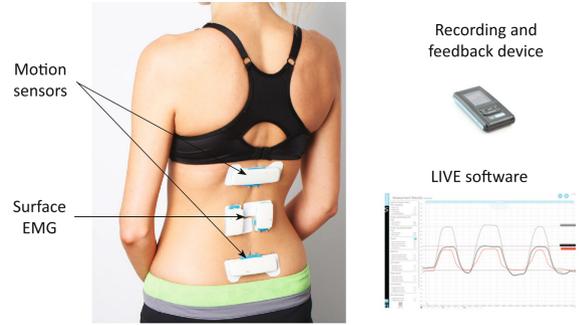


FIGURE 3. ViMove System Components [40]

Based on our experimental evaluation, this rule-based method can improve all the performance metrics, in terms of accuracy, precision and F1 score, in on-device LPMs detection. Details of the evaluation are illustrated in Section V. The serial numbers in Figure 1. indicates how this rule-based reconstruction method works on detected lumbar-pelvic movements sequence.

IV. EXPERIMENT DESIGN

This section depicts the design of the experiments which evaluate the performance of the proposed frameworks on LPMs detection and classification.

A. EXPERIMENTAL HARDWARE SETTINGS

Sensor preparations: This study used ViMove system as the data collection interface. As shown in Figure 3, ViMove system consists of four parts: motion sensors (The upper one is located at L1 and the lower one is located at PSIS), surface EMG sensors (The left one and right one are located on each side of the spine around L3), a recording and feedback device and a PC LIVE monitoring software. Each Motion sensor contains one IMU, which includes one accelerometer, one gyroscope and one magnetometer. ViMove system also provides low back fitting templates, which can be used to help users to easily attach the sensors on human back. The templates are designed based on human heights. In the meantime, to avoid the potential soft tissue artefact (STA), this study used DAP-M adhesive material, provided by ViMove, to attach the sensor on human skin. This type of material can last 24 hours for providing accurate measurement. This experiments only focus on monitoring LPMs, so the surface EMG sensors are not used. The sample rate of ViMove is 20Hz.

PC platform settings: This work used the PC platform to train the DNN models. Tensorflow was used as backend because it provided APIs and plugins to build and convert the models into tflite graphs. The generated tflite graphs could be running on mobile devices directly. On PC platform, the models were saved in HDF5 format. The proposed post-detection rule-based reasoning algorithm was also implemented on the PC platform for performance evaluation. The

TABLE 1. PC/Mobile Platform Specifications

	PC	Mobile Device
CPU	Intel Core i7-6700HQ@2.60GHz	Qualcomm Snapdragon 835 / Octa-core (4 × 2.35 GHz, 4 × 1.9 GHz) Kryo
RAM	16Gb 2133MHz SODIMM	4Gb LPDDR4X
GPU	External GPU: RTX 2070 / 8GB GDDR6 Connected by Thunderbolt 3 operating @ 40Gbps	Adreno 540
OS	Windows 10	Android 10
Battery	N/A	3520mAh

TABLE 2. Participants' Demographics

ID	Age(yrs)	Height(cm)	Weight(kg)	Gender
1	28	160	54	Female
2	27	159	50	Female
3	20	175	85	Male
4	25	162	50	Female
5	28	176	100	Male
6	29	158	50	Female
7	23	165	47	Female
8	26	164	70	Female
9	31	183	80	Male
10	28	166	62	Female
11	31	183	120	Male
12	25	173	50	Female
Max/Min	31/20	183/159	120/47	N/A
Summary (avg/std)	26.8/3.2	168.7/9.0	68.2/23.6	4 Male and 8 Female

avg = average, std = stand deviation.

detailed specifications of the PC system are described in Table 1.

Mobile platform settings: This study also evaluated and compared the performance of the proposed framework with the DNN models on the mobile platform. Google Pixel 2 XL was used as the mobile platform test device. All the modules, which were illustrated in Section III, were implemented and evaluated on this device. The detailed specifications of the mobile device are also illustrated in Table 1.

B. EXPERIMENTAL PROTOCOLS

To evaluate our proposed approach, a set of experiments were designed to simulate clinical-level LBP patients' rehabilitation progress assessment [41]. 12 participants were recruited for this experiment. All participants completed the entire experiment. The details of each participant's demographics are shown in Table 2.

Participants were instructed to perform six LPMs in sequence, including flexion, extension, left lateral flexion, right lateral flexion, left rotation and right rotation. The instructor was trained by musculoskeletal medical experts. Each participant performed each type of LPM for 15 times before starting another LPM. The flexion and extension refer to individuals bend or extend their torso forward or backward within the sagittal plane. The lateral flexions refer to individuals perform side flexion (i.e., left and right) within the coronal plane. The rotations refer to individuals rotate their torso (towards left and right) while the pelvic area facing forward. This

type of LPM is performed within the transverse plane. After performing each LPM, participants had to return to their static position (i.e., standing in a relaxed manner) for a few seconds before continuing to perform the next movement. Participants could choose to withdraw or stop to rest at any time during the experiments. In order to make the experiment comparable to real life scenario, participants are instructed to perform the LPMs at any speed with any bending or turning angles. This study has been approved by Monash University Human Research Ethics Committee.

C. DATA PROCESSING

All data was collected from ViMove system. The ViMove PC software has already applied a zero-phase, second-order Butterworth filter, and cut-off frequency of 5Hz to eliminate the noise of accelerometer and gyroscope readings. On top of that, we added another low pass Butterworth filter with a corner frequency of 0.3Hz on accelerometer readings to reduce the gravity acceleration signal interference and obtain the linear acceleration readings (also known as body acceleration signal in other literature [42]).

In order to generate DNN processable data, we first needed to slice the raw data into data chunks by using the suitable combination of sliding window size and overlap rate. Some studies suggest setting the overlap rate to 90% [39] can increase the accuracy of the models, while other studies suggest using 50% [28]. In order to determine the suitable choices of these two variables for different models, the sliding window size and overlap rate comparison experiment was first conducted on the PC platform with the three standard LSTM-based DNN models (Vanilla models).

In this experiment, 15 datasets were generated from the raw data files by using 15 combinations of five sliding window sizes (e.g., 1 sec, 1.5 sec, 2 sec, 2.5 sec and 3 sec) and three overlap rates (e.g., 30%, 50% and 70%). DNN models are stochastic models. They use randomness while being fit on a dataset, such as random initial weights and random shuffling of data during each training epoch and stochastic gradient descent [43]. This may result in different model performance after each training. Therefore, the evaluation experiments were repeated for five times, and the mean accuracy (grand mean) and standard error of the model on test datasets were calculated for the comparison.

As shown in Table 3, the most suitable sliding window size was 3.0 seconds because it produced the highest mean accuracy for all the models. However, the most suitable overlap rates of these three models were not consistent (50% for CNN-LSTM and LSTM, and 70% for Bidir-LSTM), if the highest mean accuracy was used as the only metric. Hence, the standard error was taken into considerations. We noticed that the lower the standard error, the more stable the model performance. According to this principle, 50% overlap rate seemed to be the most suitable choice for this task because all these three models produced a relatively lower standard error and higher mean accuracy with 50% overlap rate. Therefore,

TABLE 3. Overlap Rate and Sliding Window Size Comparison

			CNN-LSTM (Vanilla)	LSTM	Bidir-LSTM	
Overlap Rate	30%	Sliding Window Size (Second)	1.0	78.15% ±1.01	78.67% ±1.03	77.62% ±1.33
			1.5	79.47% ±0.96	81.29% ±0.87	79.90% ±1.46
			2.0	82.79% ±0.83	83.45% ±0.40	82.12% ±1.20
			2.5	83.47% ±0.67	84.07% ±0.71	83.53% ±0.51
			3.0	84.36% ±0.62	83.71% ±1.78	84.39% ±0.74
			3.0	84.56% ±0.58*	85.23% ±0.36*	84.79% ±0.44**
	50%	Sliding Window Size (Second)	1.0	77.93% ±0.99	78.57% ±0.42	77.43% ±0.61
			1.5	80.31% ±0.98	81.13% ±0.87	80.46% ±0.78
			2.0	82.19% ±0.81	83.18% ±0.70	82.30% ±0.67
			2.5	84.19% ±0.48	84.08% ±0.37	84.29% ±0.36
			3.0	84.56% ±0.58*	85.23% ±0.36*	84.79% ±0.44**
			3.0	84.56% ±0.58*	85.23% ±0.36*	84.79% ±0.44**
	70%	Sliding Window Size (Second)	1.0	76.82% ±1.77	76.41% ±1.85	76.31% ±1.62
			1.5	78.32% ±1.23	81.03% ±0.63	79.52% ±1.30
			2.0	81.36% ±1.41	81.42% ±1.11	81.28% ±1.51
			2.5	83.89% ±1.11	84.03% ±0.33	83.78% ±1.10
			3.0	84.33% ±0.89	85.17% ±0.79	85.03% ±1.00*
			3.0	84.56% ±0.58*	85.23% ±0.36*	84.79% ±0.44**

The bolded line texts refer to the selected combination of overlap rate and sliding window size, and their corresponding accuracies. The highest values are marked with single asterisk and the second highest value of Bidir-LSTM are marked with two asterisks since it has been selected.

3 seconds as the sliding windows size and 50% as the overlap rate were chosen in this study.

Unlike the other movement detection studies [9], [44], this research used the majority voting scheme to label the data [45]. This is because other low-level movement studies instruct the participant to finish one movement “B” within a time frame and label the entire data chunk in this time frame as “B” without considering the actual time consumed for performing this movement [9]. However, our work allowed participants to perform each LPM at any speed, this led to various time consumptions for different participant performing different LPMs. According to our statistics, the shortest time for participants performing single LPM was under two seconds, while the longest time could be 15 seconds. Therefore, the majority voting scheme was the suitable method for this work. Altogether, there were seven labels of the data, including static (0), flexion (1), left lateral flexion (2), right lateral flexion (3), left rotation (4), right rotation (5) and extension (6).

V. PERFORMANCE EVALUATION

In this section, we discuss a set of experimental evaluation that was conducted to compare the performance of proposed model and baseline models as well as the rule-based recon-

TABLE 4. Single IMU Setting Data VS Dual IMUs Setting

	IMU Settings	Accuracy	Precision	F1-Score
CNN-LSTM	Single IMU	83.07%	82.11%	82.31%
	Dual IMU	85.81%	85.04%	85.09%
LSTM	Single IMU	82.75%	81.94%	82.17%
	Dual IMU	84.72%	83.66%	83.84%
Bidir-LSTM	Single IMU	80.84%	80.12%	80.33%
	Dual IMU	85.27%	84.56%	84.60%

struction method on both PC and mobile platforms. All the hyperparameters of each model, which is presented in this section, were empirically optimised and the hyperparameters of the LSTM layer in each model were set identically for comparison. This study was a multinomial classification problem, so the following performance metrics were used to evaluate each model: normalised confusion matrix, accuracy, precision and F1 score [46]. All datasets were processed with window size 3 seconds and overlap rate 50%.

A. LPM DETECTION AND CLASSIFICATION ON PC PLATFORM

On the PC platform, we conducted three experimental performance evaluations. Firstly, the performance comparison of using single and dual IMU sensory data on LPM detection has been conducted to demonstrate why the dual IMUs system was chosen for this specific task (i.e., LPM detection). Secondly, a detailed comparison of the proposed model and baseline models on LPM detection performance has been demonstrated. Thirdly, the domain specified post-detection rule-based reconstruction method was applied after the detection and the corresponding performance with selected models were compared.

1) PERFORMANCE OF USING SINGLE AND DUAL IMU SENSORY DATA

According to the discussion in Section II, the dual IMUs-based sensor system is identified as the suitable choice for detecting and classifying lumbar-pelvic movement in an out-of-hospital setting. However, there is still lack of evidence that quantifies the difference of the performance of using single and dual sensory data on this specific task. Therefore, we designed this experiment to show how much the dual IMUs-based sensor system could improve the detection performance in terms of accuracy, precision and F1 score.

The performance of the lumbar-pelvic movements detection is highly related to sensor placements. ViMove is professional measurement system for this type of movements, so we used the same sensor placement based on its low back fitting template (ViMove Motion Sensors attached on L1 and L5 regions of the human back). On the other hand, we chose to attach the single IMU sensor (one of the ViMove Motion Sensors) on the L5 region of human back, because many studies and commercialised products have identified this position had a strong relationship with human low back area movements [11], [47].

As shown in Table 4, the experimental results demonstrate that dual-IMUs settings can improve the detection performance of all the three LSTM-based DNN models. However, for each model, the level of improvement is different. For instance, the Bidir-LSTM is more sensitive to different sensory data compared to the other models. It has more than 4% improvements of all the performance metrics. In contrast, LSTM and CNN-LSTM produce similar results (around 2% improvement) in terms of the performance metrics improvement level when using dual-IMU sensory data. To reveal the classification performance of each LPM category, we also compute the confusion matrix which is shown in Figure 4 (a) to (f). Comparing to the other LPMs, the confusion matrixes show that all the three models do not perform well on the left and right rotations with either a single IMU sensor setting or dual IMUs sensor setting. However, CNN-LSTM with dual IMU sensor setting can largely improve accuracy of detecting the left rotation compared to the other two models with the two sensor settings, see Figure 4 (a) and (b).

Therefore, the dual IMU sensor setting can be considered as more suitable than a single IMU sensor setting for detecting and classifying LPMs, because not only it can improve the overall performance of different DNN models but also potentially improve the detection accuracy for certain lumbar-pelvic movement category using certain DNN models.

2) PERFORMANCE OF PROPOSED MODEL AND BASELINE MODELS

To evaluate the accuracy of proposed model on this dataset, we compare our proposed CNN-LSTM model with four traditional machine learning models including (kNN, CART, SVM and Naïve Bayes) and four LSTM-based DNN models including (standard CNN-LSTM with early fusion [24], [27], standard CNN-LSTM with sensor-based late fusion [24], [27], LSTM [25] and Bidir-LSTM [28]).

According to [38], subject cross validation is a more suitable choice for the performance evaluation based on our dataset than standard k-fold cross validation. Therefore, this paper split the raw data into training, validation and testing datasets by participants. A random selection procedure was used to choose 3 participants' data as the test dataset and remaining data (9 participants) as the training and validation dataset. In order to further investigate the performance of models based on different ratios of training and validation dataset, we randomly selected 5, 7 and 9 participants' data respectively from the training and validation dataset and applied a 5-fold leave-one-subject-out-cross-validation (5-LOSOCV) on these three datasets for each type of models respectively. Each fold has been repeated for 3 times and the mean accuracy was calculated for the 5-LOSOCV. The results of 5-LOSOCV on the three selected training and validation datasets are shown in Figure 5, 6 and 7 respectively.

This experiment is designed to demonstrate the robustness of proposed model on different size and ratio of training/validation samples. It can be seen that our proposed model produces similar results regardless of the train-

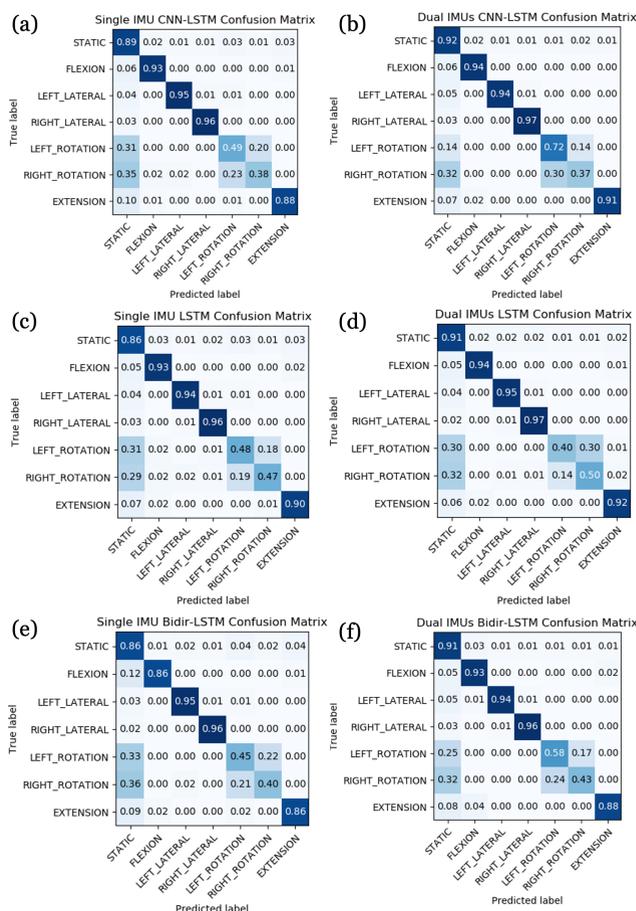


FIGURE 4. Single and Dual IMU Normalised Confusion Matrix Comparison

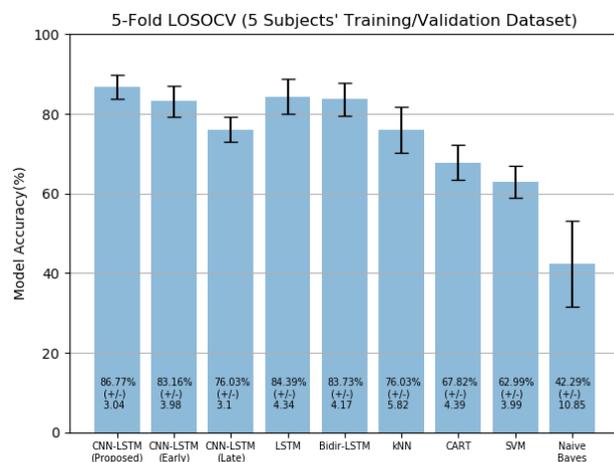


FIGURE 5. 5-Fold LOSOCV on 5 Subjects' Dataset

ing/validation dataset size. The accuracy slightly increases and the standard error decreases while the training dataset size getting larger. Although the CART shows the smallest standard error in the 9 subjects' dataset, the accuracy of the CART is significantly smaller than the proposed method.

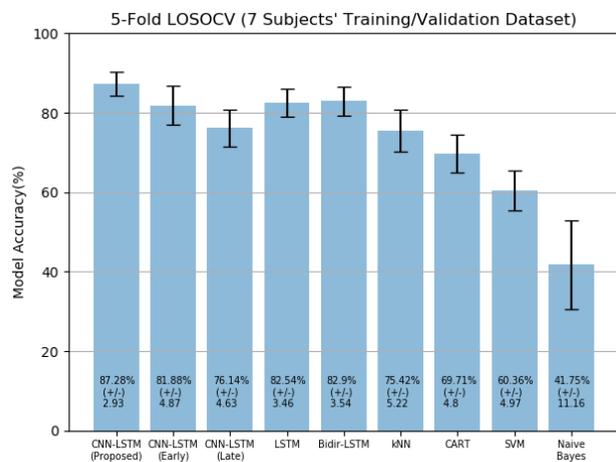


FIGURE 6. 5-Fold LOSOCV on 7 Subjects' Dataset

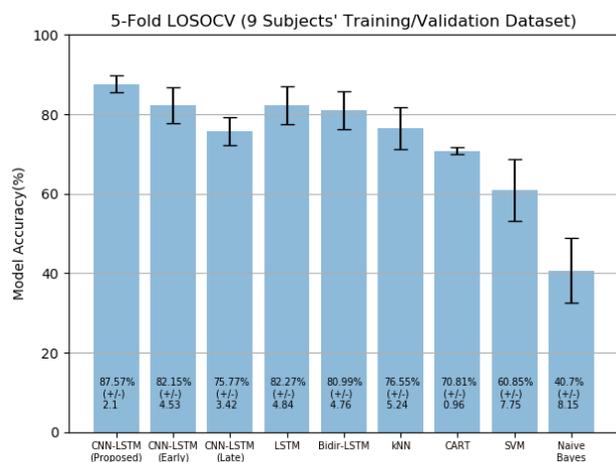


FIGURE 7. 5-Fold LOSOCV on 9 Subjects' Dataset

Additionally, the models which has the best performance on each training and validation datasets are selected to evaluate its performance on the test dataset. The results are shown in Table 5.

Additionally, the proposed CNN-LSTM model also has the highest performance on the test dataset. It improves around 10% and 3% in detection accuracy comparing to the standard CNN-LSTM and around 4% difference in terms of accuracy between our proposed model and the LSTM / Bidir-LSTM model. Since the accuracy of the conventional ML methods (kNN, CART, SVM and Naïve Bayes) and the standard CNN-LSTM model with late fusion are much lower than the remaining models, the top four models with higher accuracy (marked with *) are selected for further evaluations and comparisons.

TABLE 5. Test Datasets' Performance Comparison

	5 Subjects			7 Subjects			9 Subjects		
	Acc (%)	Pre (%)	F1 (%)	Acc (%)	Pre (%)	F1 (%)	Acc (%)	Pre (%)	F1 (%)
C-L (p)*	84.	83.	84.	86.	86.	85.	88.	87.	87.
C-L (e)*	79	96	01	18	05	98	04	70	76
C-L (l)	81.	80	80.	82.	81.	81.	84.	83.	83.
L*	34	.16	22	89	79	68	25	11	17
B-L *	71.	70.	70.	73.	72.	71.	76.	74.	75.
kN	93	82	89	22	02	98	02	97	02
N	81.	79.	79.	83.	81.	81.	84.	83.	83.
CA	25	75	82	37	82	71	72	66	84
RT	81.	79.	79.	83.	82.	82.	85.	84.	84.
SV	45	91	94	84	73	89	27	56	60
M	68.	68.	66.	73.	74.	71.	73.	73.	72.
NB	56	89	34	63	04	95	53	72	00
CA	69.	68.	67.	67.	68.	67.	71.	70.	70.
RT	20	94	82	83	19	38	09	79	37
SV	66.	71.	62.	72.	76.	70.	74.	76.	72.
M	05	01	55	99	42	59	26	66	15
NB	58.	62.	58.	57.	64.	59.	63.	69.	65.
	02	29	55	68	51	99	97	17	58

C-L = CNN-LSTM, (p) = proposed, (e) = with early fusion; (l) = with late fusion, L = LSTM, B-L = Bidir-LSTM, NB = Naïve Bayes, Acc = Accuracy; Pre = Precision; F1 = F1 Score.

3) PERFORMANCE OF DOMAIN SPECIFIC POST-DETECTION RULE-BASED RECONSTRUCTION METHOD

As discussed in Section III, the rule-based reconstruction method uses prior knowledge to rectify the detection outputs of the models and improve the overall performance. To evaluate the performance of the proposed reconstruction rules for each model, we applied these rules to the outputs of the four models and compared the change of performance metrics between the original detection outputs and reconstructed outputs.

The results are shown in Table 6 and the confusion matrices of the four models with the rule-based method are shown in Figure 8 (a) to (h). It can be seen that the rule-based method can improve all performance metrics of all the models, in terms of accuracy, precision and F1 score. Note that the detection errors are different due to the fact that different models could generate different abstract features and DNN models are stochastic models. Therefore, we acknowledge that the performance of the proposed rule-based method may produce different results with different DNN models. Compared to the performance of the proposed CNN-LSTM, LSTM and Bidir-LSTM have slightly better performance improvement after adding the rule-based method. This indicates that the method has a better performance for models with lower accuracy. This feature is important for on-device inference because after the performance of the models will drop to some extent after conversion. The details are illustrated in the next sub-section.

By further analysis of the confusion matrix, we also found that a number of rotations had been misclassified as static position. This is because the data was labelled based on video recordings, and the identification of the start and end of the transition for each rotation from a static position might have

TABLE 6. With Rule-Based Method VS Without Rule-Based Method (PC)

	Settings	Accuracy	Precision	F1-Score
CNN-LSTM (Proposed)	No-Rule	88.04%	87.70%	87.76%
	Rule	88.37%	87.92%	87.81%
CNN-LSTM (Early Fusion)	No-Rule	84.25%	83.11%	83.17%
	Rule	86.10%	85.41%	85.34%
LSTM	No-Rule	84.72%	83.66%	83.84%
	Rule	86.14%	85.30%	85.05%
Bidir-LSTM	No-Rule	85.27%	84.56%	84.60%
	Rule	86.22%	85.54%	85.37%

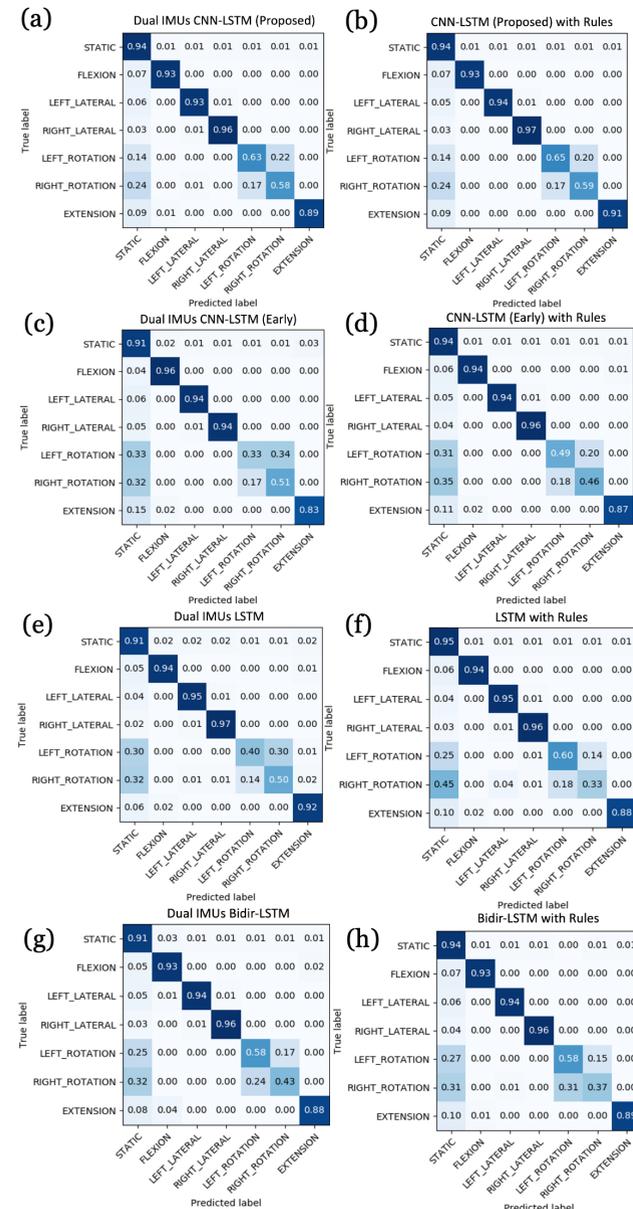


FIGURE 8. Normalised Confusion Matrix for PC Platform Evaluations (With and Without Rules)

contained errors. Also, the body rotation was mainly detected by the gyroscope. The measurement error may increase when the gyroscope moves too slowly.

B. LPM DETECTION AND CLASSIFICATION ON MOBILE PLATFORM

In this sub-section, we describe LPMs detection on a mobile platform. We first converted the previously trained models into TFLite graphs which were designed for limited computational power devices such as mobile devices [48]. Then we compared the LPMs detection and classification performance, in terms of accuracy, precision, F1-score, execution time and power consumption, of these converted models as well as the proposed rule-based reconstruction method.

TFLite library is part of Google’s Tensorflow [49]. It offers a significantly reduced binary size and kernels optimized for on-device inference [29]. TFLite is also supported by Android Neural Networks API (NNAPI) which can utilise the on-device AI hardware acceleration resources (such as the mobile GPUs) directly to improve the inference performance. However, TFLite is still lacking the support of some vital deep learning operators, especially for RNNs such as LSTM [50]. Therefore, this work used the TensorFlow Lite plugin, which contained the standard Tensorflow operators [29], to implement and convert these selected DNN models for LPM detection and classification on the mobile devices.

To thoroughly investigate the on-device performance of these three models as well as the proposed rule-based method, we separated the evaluation of the performance into two parts: 1) selected DNN models’ performance comparison; 2) on-device processing performance comparison.

1) PERFORMANCE OF SELECTED DNN MODELS WITH AND WITHOUT RULE-BASED RECONSTRUCTION METHOD

This sub-section compares the performance metrics, including accuracy, precision and F1 score, of the four converted DNN models with and without the proposed rule-based reconstruction method.

As shown in Table 7, the performance metrics of all the models dropped after the conversion. The Bidir-LSTM has the sharpest drop while the LSTM has the smallest decrease level. CNN-LSTM based models also has a larger decrease ($\approx 3\%$) comparing to the LSTM. This is because the quantised inference was used in this work instead of the floating-point inference [29]. Quantised inference needs to convert the model from a 16-bit floating point type to int-8 format to reduce the size and random-access memory (RAM) consumption by a factor of 4 [29]. This mechanism may reduce the execution time by 2 or 3 times. In addition, integer computations can significantly decrease the energy consumption on most of the mobile devices [49]. In order to make the models suitable for the general mobile devices, we chose to use this type of inference. However, the bit-width network weights reduction may lead to different level of accuracy losses for different tasks [48]. This could explain why all the performance metrics of the models dropped after the conversion.

After applying the rule-based method to improve the detection accuracy, the performance results for all the models

TABLE 7. Model Performance Comparison on Mobile Device

	Settings	Accuracy	Precision	F1-Score
	CNN -LSTM (Proposed)	PC	88.04%	87.70%
Mobile		85.11%	84.87%	84.52%
Mobile (Rules)		86.23%	85.77%	85.92%
CNN -LSTM (Early Fusion)	PC	84.25%	83.11%	83.17%
	Mobile	81.56%	81.52%	81.17%
	Mobile (Rules)	83.29%	83.44%	82.91%
LSTM	PC	84.72%	83.66%	83.84%
	Mobile	84.01%	82.97%	83.21%
	Mobile (Rules)	85.17%	84.20%	84.38%
Bidir -LSTM	PC	85.27%	84.56%	84.60%
	Mobile	36.88%	42.98%	32.39%
	Mobile (Rules)	44.53%	50.18%	39.98%

PC = PC Platform, Mobile = Mobile Platform, Mobile (Ru) = Mobile Platform with Rule-based Method.

TABLE 8. Mobile Platform Processing Performance Comparison

	Settings	Average Processing Time (ms)	Processing Power Consumption (3hrs/With Rules)
CNN-LSTM (Proposed)	No-Rule	2.74 ±0.79	1297.9 mAh
	Rule	9.53 ±1.89	
CNN-LSTM (Early Fusion)	No-Rule	1.89 ±0.88	1213.6 mAh
	Rule	4.21 ±1.75	
LSTM	No-Rule	4.14 ±0.64	1372.8 mAh
	Rule	12.91 ±2.23	
Bidir-LSTM	No-Rule	16.91 ±1.52	1443.3 mAh
	Rule	54.81 ±14.76	

show different levels of improvement. The LSTM has the smallest level of improvement while the Bidir-LSTM has the largest level of improvement. However, the performance of the Bidir-LSTM is still the worst because of the enormous drop during the conversion and the LSTM even performs better compared to PC platform (without rule-based method). It can also be found that the rule-based method helps the proposed CNN-LSTM to achieve the best on-device performance of the selected models.

2) PERFORMANCE OF MOBILE PLATFORM PROCESSING

In this sub-section, the on-device processing performance of each model with and without the rule-based method, in terms of execution time and power consumption, are compared.

The on-device processing evaluation only compares the inferring time and signal pre-processing time of each model rather than the entire data transmitting and execution time, because this study uses the ViMove PC software collected data. The experimental evaluation procedures are as follows. First, we randomly selected 45 mins data from the test dataset because real-life low back movement assessment takes no more than 45 mins [41]. Second, the data is stored into the local storage on the smartphone and ran the on-device inference for all the data. To simulate the real-time processing scenario, each data line was read per 50 ms (equivalent to the sampling rate 20Hz). Third, we calculated the average processing time of each data chunk.

The entire process ran two times with each model, with

and without rule-based method respectively. The results are shown in Table 8. It can be seen that the standard early fusion CNN-LSTM has the quickest processing time (including inferring time and signal processing time) among the four models. After applying the rule-based reconstruction method, the standard early fusion CNN-LSTM is still the quickest one. On the other hand, due to the computation complexity of the Bidir-LSTM, it has the longest processing time, which is more than 4 times of the LSTM and the proposed CNN-LSTM process time, and 8 times of the stand early fusion CNN-LSTM.

The power consumption performance evaluation test is designed to compare the energy efficiency of using these four models on the mobile platform. In this evaluation, we directly used the models with the rule-based method because one of the goals of this study was to evaluate the performance of the proposed framework on a mobile platform and the rule-based model is part of the framework. Each model ran consistently for 3hrs on the mobile phone for 5 times. The phone was fully charged (100%) at the beginning and on aeroplane model with the screen off and all background tasks were killed during each test. The calculation formula (6) of the power consumption is shown as follows.

$$Power\ Consumption = (100\% - remaining\ power) \times battery\ capacity \quad (6)$$

The results of the average power consumption for each model are also shown in Table 8. All the models have a similar power consumption level. The most energy-efficient one is the standard early fusion CNN-LSTM, which consumes average 1213.6 mAh. The worst one is Bidir-LSTM which consumes average 1443.3 mAh. We acknowledge that different smartphones may result in different power consumption performances.

By considering the trade-off between on-device processing speed, power consumption and detection accuracy, the overall results show the proposed CNN-LSTM is the most energy efficient for on-device real-time LPM detection and classification, comparing to other selected DNN models. Although the standard early fusion CNN-LSTM model has a quicker on-device execution speed and less power consumption than other models, its accuracy as well as other performance metrics are significantly lower than the proposed CNN-LSTM model. On the other hand, the Bidir-LSTM suffers from a huge accuracy loss when applied on a mobile platform, which makes it not suitable for any mobile platform computing at this stage.

VI. CONCLUSION AND FUTURE WORK

This paper presented a DNN based approach to detect LPMs, including flexion, lateral flexion, rotation and extension, locally on-device, where the data was collected from a clinically validated dual IMUs sensor system. We also proposed an enhanced adapted CNN-LSTM model which utilises domain adaptation for feature augmentation and

global temporal-wise attention mechanism for additional context information processing. Additionally, a rule-based reconstruction method was developed and integrated into the approach to increase the accuracy of the detection performance. Our experimental evaluation results demonstrated that adding domain adaptation technique and attention mechanism can significantly improve the performance of CNN-LSTM on our dataset. Additionally, the proposed model is suitable for processing multiple IMUs sensory data and the presented approach has a promising real-time on-device detection performance including efficient power consumption rate (sufficient battery life for 8 hours monitoring in this case) and acceptable LPM detection accuracy based on the medical experts' feedback.

In the future work, we plan to conduct more experiments with different real daily living scenarios (such as cleaning floor and washing dishes) to collect a wide range of real data for establishing a robust DNN model to detect LPMs in human random daily activities. Other DNN models and adaptive sliding window size techniques to increase the accuracy in detecting the left/right rotation also need to be considered.

ACKNOWLEDGMENT

The authors would like to thank Dr. Ying Xu for her helpful advice on various technical issues examined in this paper.

REFERENCES

- [1] D. Steffens, M. L. Ferreira, J. Latimer, P. H. Ferreira, B. W. Koes, F. Blyth, Q. Li, and C. G. Maher, "What triggers an episode of acute low back pain? a case-crossover study," *Arthritis care & research*, vol. 67, no. 3, pp. 403–410, 2015.
- [2] P. K. D. Pramanik, B. K. Upadhyaya, S. Pal, and T. Pal, "Internet of things, smart sensors, and pervasive systems: Enabling connected and pervasive healthcare," in *Healthcare Data Analytics and Management*. Elsevier, pp. 1–58, 2019.
- [3] M. Shoaib, S. Bosch, O. D. Incel, H. Scholten, and P. J. Havinga, "Complex human activity recognition using smartphone and wrist-worn motion sensors," *Sensors*, vol. 16, no. 4, p. 426, 2016.
- [4] E. Charry, M. Umer, and S. Taylor, "Design and validation of an ambulatory inertial system for 3-d measurements of low back movements," in *Proc. Int. Conf. Intell. Sensors, Sens. Networks Inf. Process.* IEEE, pp. 58–63, 2011.
- [5] T. de Quadros, A. E. Lazzaretti, and F. K. Schneider, "A movement decomposition and machine learning-based fall detection system using wrist wearable device," *IEEE Sens. J.*, vol. 18, no. 12, pp. 5082–5089, 2018.
- [6] M. Shoaib, S. Bosch, O. D. Incel, H. Scholten, and P. J. Havinga, "A survey of online activity recognition using mobile phones," *Sensors*, vol. 15, no. 1, pp. 2059–2085, 2015.
- [7] S. Yao, S. Hu, Y. Zhao, A. Zhang, and T. Abdelzaher, "Deepsense: A unified deep learning framework for time-series mobile sensing data processing," in *Proc. World Wide Web Conf.*, pp. 351–360, 2017.
- [8] C. M. Bauer, M. Heimgartner, F. M. Rast, M. J. Ernst, S. Oetiker, and J. Kool, "Reliability of lumbar movement dysfunction tests for chronic low back pain patients," *Man. Ther.*, vol. 24, pp. 81–84, 2016.
- [9] M. Takata, Y. Nakamura, Y. Torigoe, M. Fujimoto, Y. Arakawa, and K. Yasumoto, "Strikes-thrusts activity recognition using wrist sensor towards pervasive kendo support system," in *Int. Conf. Pervasive Comput. Commun. Work. PerCom Work.*. IEEE, pp. 243–248, 2019.
- [10] D. C. Ribeiro, G. Sole, J. H. Abbott, and S. Milosavljevic, "Validity and reliability of the spineangel@ lumbo-pelvic postural monitor," *Ergonomics*, vol. 56, no. 6, pp. 977–991, 2013.
- [11] Y.-L. Kuo, P.-S. Wang, P.-Y. Ko, K.-Y. Huang, and Y.-J. Tsai, "Immediate effects of real-time postural biofeedback on spinal posture, muscle activity, and perceived pain severity in adults with neck pain," *Gait & posture*, vol. 67, pp. 187–193, 2019.
- [12] K. James, "Effects of the upright™ posture training program on spinal angles and self-esteem," M.S. thesis, Texas Tech University, 2018.
- [13] Y. Zhang, P. D. Haghighi, F. Burstein, L. W. Yap, W. Cheng, L. Yao, and F. Cicutini, "Electronic skin wearable sensors for detecting lumbar–pelvic movements," *Sensors*, vol. 20, no. 5, p. 1510, 2020.
- [14] S. Schellendorfer, M. J. Ernst, F. M. Rast, C. M. Bauer, A. Meichtry, and J. Kool, "Low back pain and postural control, effects of task difficulty on centre of pressure and spinal kinematics," *Gait & posture*, vol. 41, no. 1, pp. 112–118, 2015.
- [15] A. Singh, A. Klapper, J. Jia, A. Fidalgo, A. Tajadura-Jiménez, N. Kanakam, N. Bianchi-Berthouze, and A. Williams, "Motivating people with chronic pain to do physical activity: opportunities for technology design," in *Proc. SIGCHI Conf. Hum. factors Comput. Syst.*, pp. 2803–2812, 2014.
- [16] J. Li, Z. Wang, S. Qiu, H. Zhao, Q. Wang, D. Plettemeier, B. Liang, and X. Shi, "Using body sensor network to measure the effect of rehabilitation therapy on improvement of lower limb motor function in children with spastic diplegia," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 11, pp. 9215–9227, 2020.
- [17] S. Qiu, Z. Wang, H. Zhao, and H. Hu, "Using distributed wearable sensors to measure and evaluate human lower limb motions," *IEEE Trans. Instrum. Meas.*, vol. 65, no. 4, pp. 939–950, 2016.
- [18] D. Roetenberg, H. Luinge, and P. Slycke, "Xsens mvn: Full 6dof human motion tracking using miniature inertial sensors," *Xsens Motion Technologies BV, Tech. Rep.*, vol. 1, 2009.
- [19] G. Marta, F. Simona, C. Andrea, B. Dario, S. Stefano, V. Federico, B. Marco, B. Francesco, M. Stefano, and P. Alessandra, "Wearable biofeedback suit to promote and monitor aquatic exercises: a feasibility study," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 4, pp. 1219–1231, 2019.
- [20] Q. Yuan and I.-M. Chen, "3-d localization of human based on an inertial capture system," *IEEE Trans. Robot.*, vol. 29, no. 3, pp. 806–812, 2013.
- [21] H. L. Mjøsund, E. Boyle, P. Kjaer, R. M. Mieritz, T. Skallgård, and P. Kent, "Clinically acceptable agreement between the vimove wireless motion sensor system and the vicon motion capture system when measuring lumbar region inclination motion in the sagittal and coronal planes," *BMC musculoskelet. disord.*, vol. 18, no. 1, pp. 1–9, 2017.
- [22] L. M. Dang, K. Min, H. Wang, M. J. Piran, C. H. Lee, and H. Moon, "Sensor-based and vision-based human activity recognition: A comprehensive survey," *Pattern Recognit.*, vol. 108, p. 107561, 2020.
- [23] K. Chen, D. Zhang, L. Yao, B. Guo, Z. Yu, and Y. Liu, "Deep learning for sensor-based human activity recognition: overview, challenges and opportunities," *arXiv preprint arXiv:2001.07416*, 2020.
- [24] S. Münzner, P. Schmidt, A. Reiss, M. Hanselmann, R. Stiefelhofen, and R. Dürichen, "Cnn-based sensor fusion techniques for multimodal human activity recognition," in *Proc. Int. Symp. Wearable Comput.*, pp. 158–165, 2017.
- [25] Y. Guan and T. Plötz, "Ensembles of deep lstm learners for activity recognition using wearables," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 1, no. 2, pp. 1–28, 2017.
- [26] S. Ha and S. Choi, "Convolutional neural networks for human activity recognition using multiple accelerometer and gyroscope sensors," in *Int. Joint Conf. Neural Networks*. IEEE, pp. 381–388, 2016.
- [27] F. J. Ordóñez and D. Roggen, "Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, p. 115, 2016.
- [28] Y. Zhao, R. Yang, G. Chevalier, X. Xu, and Z. Zhang, "Deep residual bidir-lstm for human activity recognition using wearable sensors," *Math. Probl. Eng.*, vol. 2018, 2018.
- [29] A. Ignatov, R. Timofte, A. Kulik, S. Yang, K. Wang, F. Baum, M. Wu, L. Xu, and L. Van Gool, "Ai benchmark: All about deep learning on smartphones in 2019," in *IEEE/CVF Int. Conf. Comp. Vision Work. (ICCVW)*. IEEE, pp. 3617–3635, 2019.
- [30] D. Zhang, Z. Pan, J. Chen, and Y. Xie, "A new method of bio-medical image fusion with multi-scale edge," *J. Inf. & Comput. Sci.*, vol. 11, no. 16, pp. 5681–5688, 2014.
- [31] A. Ignatov, "Real-time human activity recognition from accelerometer data using convolutional neural networks," *Appl. Soft Comput.*, vol. 62, pp. 915–922, 2018.
- [32] Y. Chen and Y. Xue, "A deep learning approach to human activity recognition based on single accelerometer," in *Int. Conf. Syst. Man, Cybern.* IEEE, 2015, pp. 1488–1492.

[33] H. Daumé III, "Frustratingly easy domain adaptation," *arXiv preprint arXiv:0907.1815*, 2009.

[34] F. M. Noori, E. Garcia-Ceja, M. Z. Uddin, M. Riegler, and J. Tørresen, "Fusion of multiple representations extracted from a single sensor's data for activity recognition using cnns," in *Int. Joint Conf. Neural Networks*. IEEE, pp. 1–6, 2019.

[35] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *arXiv preprint arXiv:1508.04025*, 2015.

[36] S. Mahmud, M. Tonmoy, K. K. Bhaumik, A. Rahman, M. A. Amin, M. Shoyaib, M. A. H. Khan, and A. A. Ali, "Human activity recognition from wearable sensor data using self-attention," *arXiv preprint arXiv:2003.09018*, 2020.

[37] K. Chen, L. Yao, X. Wang, D. Zhang, T. Gu, Z. Yu, and Z. Yang, "Interpretable parallel recurrent neural networks with convolutional attentions for multi-modality activity modeling," in *Int. Joint Conf. Neural Networks*. IEEE, pp. 1–8, 2018.

[38] A. Dehghani, T. Glatard, and E. Shihab, "Subject cross validation in human activity recognition," *arXiv preprint arXiv:1904.02666*, 2019.

[39] S. Mehrang, J. Pietilä, and I. Korhonen, "An activity recognition framework deploying the random forest classifier and a single optical heart rate monitoring and triaxial accelerometer wrist-band," *Sensors*, vol. 18, no. 2, p. 613, 2018.

[40] P. Kent, R. Laird, and T. Haines, "The effect of changing movement and posture using motion-sensor biofeedback, versus guidelines-based care, on the clinical outcomes of people with sub-acute or chronic low back pain: multicentre, cluster-randomised, placebo-controlled, pilot trial," *BMC musculoskelet. disord.*, vol. 16, no. 1, pp. 1–19, 2015.

[41] J. Guzmán, R. Esmail, K. Karjalainen, A. Malmivaara, E. Irvin, and C. Bombardier, "Multidisciplinary rehabilitation for chronic low back pain: systematic review," *BMJ*, vol. 322, no. 7301, pp. 1511–1516, 2001.

[42] D. N. Tran and D. D. Phan, "Human activities recognition in android smartphone using support vector machine," in *Proc. - Int. Conf. Intell. Syst. Model. Simulat.*. IEEE, pp. 64–68, 2016.

[43] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[44] H. Zhao, S. Wang, G. Zhou, and W. Jung, "Tenniseye: tennis ball speed estimation using a racket-mounted motion sensor," in *Proc. Inf. Process. Sens. Networks*, 2019, pp. 241–252.

[45] A. Hajdu, L. Hajdu, A. Jonas, L. Kovacs, and H. Toman, "Generalizing the majority voting scheme to spatially constrained voting," *IEEE Trans. Image Process.*, vol. 22, no. 11, pp. 4182–4194, 2013.

[46] E. Beauxis-Aussalet and L. Hardman, "Simplifying the visualization of confusion matrix," *BNAIC, November*, 2014.

[47] P. Intolo, A. B. Carman, S. Milosavljevic, J. H. Abbott, and G. D. Baxter, "The spineangel@: Examining the validity and reliability of a novel clinical device for monitoring trunk motion," *Man. Ther.*, vol. 15, no. 2, pp. 160–166, 2010.

[48] C. Alippi, S. Disabato, and M. Roveri, "Moving convolutional neural networks to embedded systems: the alexnet and vgg-16 case," in *Proc. Int. Conf. Inf. Process. Sens. Networks*. IEEE, pp.212–223, 2018.

[49] M. S. Louis, Z. Azad, L. Delshadtehrani, S. Gupta, P. Warden, V. J. Reddi, and A. Joshi, "Towards deep learning using tensorflow lite on risc-v," in *Third Work. Comput. Archit. Res. with RISC-V*, vol. 1, p. 6, 2019.

[50] M. Xu, J. Liu, Y. Liu, F. X. Lin, Y. Liu, and X. Liu, "A first look at deep learning apps on smartphones," in *Proc. World Wide Web Conf.*, pp. 2125–2136, 2019.



YUXIN ZHANG (S'18–M'21) completed the Bachelor of Engineering degree in Automation and the Master of Computer Technology degree from the Southeast University, Nanjing, China. He received the Master of Information Technology Systems degree from Monash University, Melbourne, Australia. He is currently pursuing Ph.D. degree with the Faculty of Information Technology at Monash University and holding a Post Doctoral Position at Monash University researching

sensor technology in healthcare. His research interests are in the area of smart sensors and sensing technology for digital health, and edge AI.



PARI DELIR HAGHIGHI received her PhD in Computing from Monash University in 2010. She is currently a Senior Lecturer at the Department of Human-Centred Computing, Faculty of IT, Monash University. Her research interests include context-aware and ubiquitous computing, mobile sensing, decision support systems, IoT systems, and digital health.



FRADA BURSTEIN holds PhD in decision support systems from the Georgian Academy of Sciences. She is Adjunct Professor at the Digital Equity and Digital Transformation Group at the Department of Human-Centred Computing, Monash University, Australia. Prof Burstein has a strong international reputation in decision support, health informatics and knowledge management. She published more than 200 articles which appear in such journals as Decision Support Systems, IT and

People, Organisational Computing and Electronic Commerce, Knowledge Management and Knowledge Management Research and Practice, Journal of IT, Communication of AIS and others. She is Distinguished Member of the Association for Information Systems and a Fellow of Australian Computer Society.



LINA YAO (M'14) received the master's and Ph.D. degrees from the University of Auckland (UoA), Auckland, New Zealand, in 2010 and 2014, respectively. She is currently an Associate Professor with the School of Computer Science and Engineering, University of New South Wales, Sydney, NSW, Australia. Her current research interests include data mining and machine learning, recommender systems, and human activity recognition.



FLAVIA CICUTTINI graduated in medicine from Monash University in 1982 and completed a PhD at the University of Melbourne in 1993. Prof Cicuttini then completed a MSc at the University of London, looking at risk factors for osteoarthritis and completed further study at the London School of Hygiene and Tropical Medicine in 1996. Current research includes using Magnetic Resonance Imaging to understand factors that affect joint cartilage in healthy and diseased states.

...