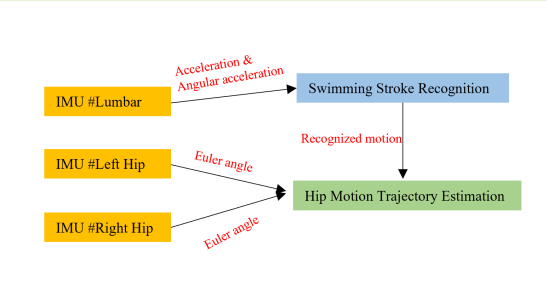
A Deep Learning Control Strategy of IMU-based Joint Angle Estimation for Hip Power-Assisted Swimming Exoskeleton

Longwen Chen, Dean Hu*

***Abstract*— Wearable exoskeleton techniques are becoming mature and wildly used in many areas. However, the biggest challenge lies in that the control system should recognize and follow the wearer's motion correctly and quickly. In this study, we propose a deep learning control strategy utilizing inertial measurement units (IMU) for hip power-assisted swimming exoskeleton. The control strategy includes two steps: Step 1, the swimming stroke is recognized by a deep convolutional neural and bidirectional long short-term memory network (DCNN-BiLSTM); Step 2, the hip joint angles are estimated with BiLSTM network belonging to the recognized motion to predict the hip trajectory. The dataset of motion recognition and estimation of four swimming strokes are collected by placing IMUs on swimmers' back and thighs. We conduct offline and online testing of control strategy for accuracy and robustness validation. During offline testing, we achieve an accuracy more than 96% of motion recognition and root mean square error (RMSE) less than 1.2° of hip joint angle estimation, outperforming 2.76% of accuracy and 0.09° of RMSE compared with those of ELM or CNN-GRU. During online testing, the pre-trained networks are transplanted into a Raspberry Pi 4B and achieve 8.47ms for conducting one motion recognition and 6.72ms for one hip joint angle estimation on average, which are far less than 300ms of delayed sensations between the action of exoskeleton and human, while keeping a satisfying recognition accuracy as well. The experiment results show that the accuracy and robustness of proposed control strategy are stable and feasible for the application to exoskeletons.**



***Index Terms*— Motion recognition; Motion trajectory estimation; Deep learning; Hip power-assisted swimming exoskeleton; Control strategy.**

I. Introduction

WEARABLE exoskeletons have been applied to many fields such as power assistance, rehabilitation training, agriculture, military [1]–[4] etc. Power-assisted exoskeletons can repeatedly and correctly lead the wearer's limbs through a pre-defined motion trajectory and cut down the joint pressure and energy cost of the wearer [5]. They can even help the wearer conduct tasks far beyond their ability. Swimming, as one of the most popular forms of exercise among people, is acknowledged as an excellent but tiring aerobic exercise. Four competitive swimming stroke (i.e. backstroke, breaststroke, butterfly, front crawl) all depend on thigh strength, which mainly generates from hip joint of each leg, to provide the propelling force of the swimmer [6]. With the help of hip power-assisted exoskeletons, the swimmer's hip fatigue can be greatly relieved.

The most difficult technique of power-assisted exoskeleton lies in that it should be able to recognize and follow the wearer's motion and trajectory while maintaining a satisfying accuracy and robustness. Some phase detection based control strategies have been proposed and applied on exoskeletons. Our previous work [7] presents an underwater power-assisted device controlled by the detected frequency of forearm swings, which is collected by an IMU on the wearer's forearm, to adjust the propelling force generated by two thrusters in order to produce power assistance for the wearer. Wang et al. [8] presents a swimming assistance soft exoskeleton based on breaststroke motions. They applied state transition based phase detection as the control strategy of their exoskeleton to recognize three motion phases (glide, recovery and sweep) during breaststroke. Yu et al. [9] presented a torque tracking strategy for squatting assistance control system applied on a lower limb exoskeleton. Their control strategy detects lifting phase during squatting with inertial measurement units (IMUs) and generates a calculated assistive profile to the wearer's hip joint. Natali et al. [10] shows a control strategy that using the signal output of the shoe insole to determine three gait phases (early stance, mid stance and swing) in order to realize motion recognition of proposed soft assistive lower limb exoskeleton. Phase detection based control strategy can overcome the above

Key Laboratory of Advanced Design and Simulation Techniques for Special Equipment, Ministry of Education, Hunan University, Changsha 410082, P. R. China (e-mail: author@boulder.nist.gov).
*Corresponding author: (e-mail: hudean@hnu.edu.cn)(Dean Hu).

mentioned difficulty. However, the major limitation lies in that it can only recognize one specific motion. During swimming, for example, the swimming stroke of the swimmer can be switched randomly. Phase detection based control strategy cannot meet the requirement of user's motion switches.

More and more studies related to motion recognition or estimation start to apply machine/deep learning networks (ML/DL) to conduct corresponding tasks. The continuous videos, pictures, IMU or EMG signals are used as input of ML/DL networks to recognize/predict human motions [1], [12], [13] For example, our previous work [14] presents a swimming stroke recognition system based on two IMUs placed on the lumbar and back of the swimmer, utilizing a deep convolutional neural and bidirectional long short-term memory network (DCNN-BiLSTM). Vijay et al. has a research on human activity recognition such as walking activities recognition [15]–[18], gaits trajectories associated with different joint angle trajectories, walking instance and speeds [19]–[21], upper limb exercises of post-stroke rehabilitation patients [22], and fall detection [23] with hybrid DL techniques such as extreme learning machine (ELM) or CNN-RNN (e.g. inception based CNN-GRU) networks, and receives impressing and convincing experiment results. Yijian Chen et al. [24] conducts an IMU-based lower limb motion trajectory estimation with a graph convolution network. Siyu Chen et al. [25] uses a single IMU for real-time walking gait estimation of construction workers. Sung et al. [26] utilizing a single IMU to predict multi-joint angles during walking based on LSTM network, etc. ML/DL based control strategy can greatly solve the above mentioned limitation. Ren et al. [27] proposed a deep learning based control strategy on an upper limb rehabilitation exoskeleton robot for motion prediction in order to produce torque assistance for the wearer. Zhu et al. [28] presents an IMU based motion recognition for soft exoskeleton through DNN for locomotion identification to provide different assistance mode for the user. Shinnosuke et al. [29] places motion sensors on the wearer's torso and legs to estimate his/her motion to enable the powered exoskeleton to assist with the estimated motion through KNN. However, not only above mentioned studies, but also other studies related to wearable exoskeletons in [30]–[35] all focus on the land activities like walking, running or gesture recognition of the user. Control strategies of power-assisted exoskeleton aiming at assisting underwater activities such as swimming are hardly to discover.

Therefore, based on our previous work, in this study, we propose a DL control strategy of IMU-based joint angle estimation for hip power-assisted swimming exoskeleton. To realize the proposed strategy, the lumbar and each thigh of the swimmer are placed with one IMU respectively for real-time kinematic data collection. The acceleration and angular acceleration collected from the lumbar IMU are used as the dataset for motion recognition, while the hip joint angles calculated from quaternions of each hip IMU are used to conduct motion estimation. The major contributions of this study are as follows:

(1). A versatile kinematic dataset of four swimming strokes (collected by three IMUs and contains acceleration and angular acceleration of the lumbar, joint angles of hips) of three well-trained swimmers is established;

(2). A DCNN-BiLSTM classification network for swimming stroke recognition, and four BiLSTM regression networks of four swimming strokes for hip angle estimation are proposed, and the performance of proposed DL networks conducting corresponding task in offline testing are proved to be acceptable, outperforming 2.76% of accuracy and 0.09° of RMSE compared with those of ELM or CNN-GRU;

(3). The proposed DL control strategy is transplanted into a Raspberry Pi 4B for online performance evaluation, and achieves averagely 8.47ms for one motion recognition while keeping a satisfying recognition accuracy, and 6.72ms on average for one hip joint angle estimation, which are far less than 300ms of delayed sensations between the action of exoskeleton and human.

The innovations of this work mainly lie in the application field and platform of motion recognition and estimation utilizing DL networks. When applying our control strategy to underwater power-assisted exoskeletons, conducting motion recognition and estimation is helpful for the user's real-time motion detection and following for assistance control, while running on Raspberry Pi indicates that the exoskeleton does not need to be connected to a PC, which helps reduce the size of the exoskeleton without limiting the motion range of the user. Furthermore, the hardware of our control strategy containing only three IMUs, a Raspberry Pi and a battery makes it easy to be applied to exoskeletons. Finally, our study contributes to the research of underwater human motion and underwater power-assisted exoskeleton in some ways.

The structure of this paper is listed as follows: Section II illustrates the experiment of data collection and data preprocess of this study; Section III explains the DL control strategy in detail, including the overview of control strategy, related DL theory and DL networks applied for motion recognition and estimation; Section IV presents the offline experiment results of swimming stroke recognition and hip motion estimation utilizing proposed DL methods, compares the corresponding results with other DL methods, and provides online experiment results of the robustness of conducting motion recognition and estimation; Section V demonstrates the discussion and conclusion of our work.

II. Data Acquisition and Preprocess

A. Experiment for Data Acquisition

Three well-trained swimmers in Hunan Provincial Swimming Team (Changsha, Hunan Province, China) are invited for our data acquisition experiments. All of them are informed with related matters needing attention and are voluntarily participate in the experiment. Tab.I illustrates the basic information of our invited subjects.

Three IMUs (Xsens DOT, Netherlands) are placed on the lumbar and thighs of each subject according to the ISB recommendation of joint coordinate definitions [36], [37]. Each sensor has a size of $12\times24\times5\text{mm}^3$ and a weight of 7g only. Before the data collection, each IMU's axes are calibrated by performing static calibration on standing pose

TABLE I: The basic information of swimmers.

Subject	Gender	Age	Height	Weight
1	Male	25	1.81m	78kg
2	Female	28	1.67m	51kg
3	Male	22	1.71m	74kg

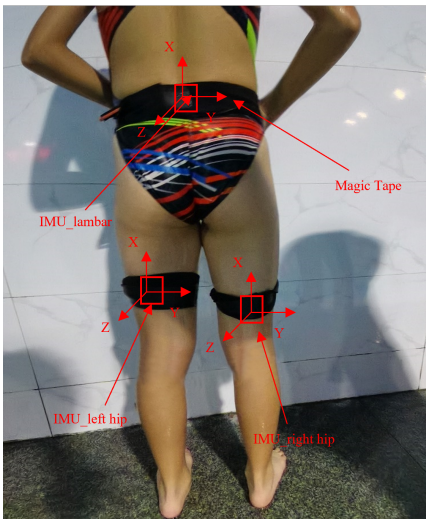


Fig. 1: The placement and coordinate definition of each IMU.

(i.e. A pose). The experiment takes place in a standard 50m swimming pool. IMUs are wirelessly connected to a cellphone and are decided the beginning and ending of data collection by a Xsens DOT APP. IMUs are attached to the swimmer's body by magic tapes to prevent fall off, just as shown in Fig. 1 below.

During data acquisition experiment shown in Fig. 2, all the subjects are asked to swim in the order of backstroke, breaststroke, butterfly and front crawl for 50m respectively for 4 groups at their 70% effect. The interval between each group for a short break is 3 mins and that of each stroke in a group is 1 min in order to prevent fatigue of the swimmers. The sampling frequency of IMUs are 120Hz, and the collected data can be recorded and stored in the designed motion order in the memory card of sensor for offline data processing. The computer for offline training and testing is equipped with Intel i5-12490F CPU, RTX 6050 GPU and 16GB RAM. All the DL networks used in this study are coded based on Pytorch.

B. Data Preprocess

The collected data are handled by a Kalman filtering algorithm to cut down nonlinear noise. Due to the sample frequency 120Hz during data acquisition, a sliding window with length of 120ms and overlap with 60ms shown in Fig. 3 below is applied to segment the collected data before being set to conduct motion recognition or estimation in order to speed up the training speed.

For motion recognition, similar to our previous work, "one-hot encoding" method, which can be explained in Tab. II below, is utilized for data labeling during preprocess. Convolutional layer can deal with automatically learning and sorting

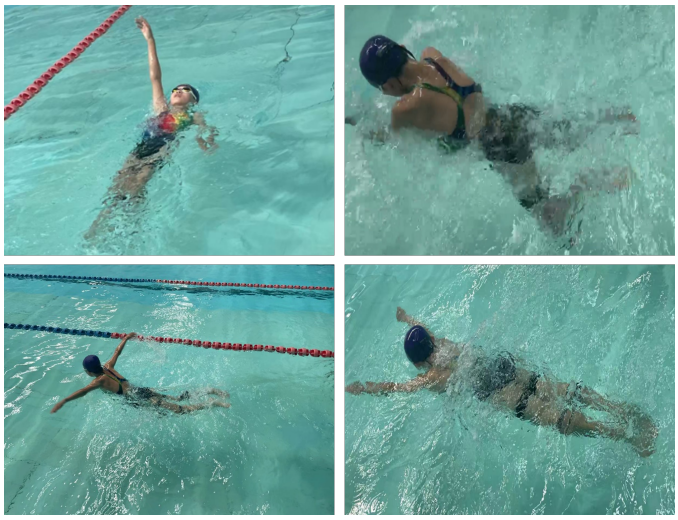


Fig. 2: The data acquisition experiment. For each swimmer, each motion group is conducted in the order of backstroke, breaststroke, butterfly and front crawl for 50m respectively. In total 4 motion groups are taken place on each swimmer.

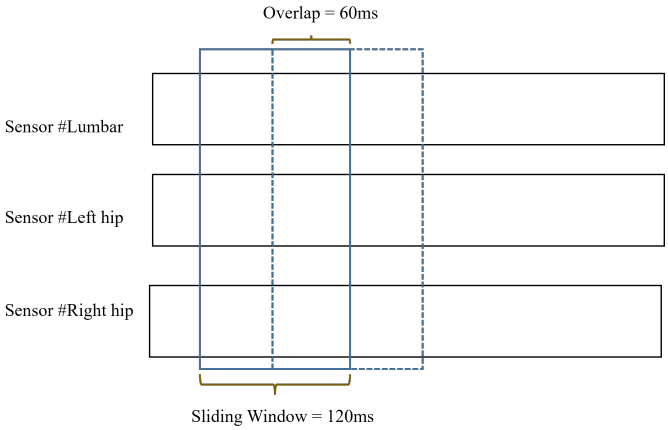


Fig. 3: The definition of sliding window.

features from raw data [38], [39]. Therefore, the segmented lumbar IMU data are used as the input of DCNN-BiLSTM directly without normalized to automatically conduct feature extraction. Data belonging to the same motion label are concatenated in order to enlarge our training set.

Euler angles calculated by the quaternion of IMUs placed on the thigh can be used to express the angles of freedom belonging to one joint. According to the joint coordination shown in Fig. 1, pitch angles can be chosen to directly reflect the hip joint angles, which is calculated by Eq. (1)

$$\varphi = \arctan \frac{2q_2q_3 + 2q_0q_1}{q_0^2 - q_1^2 - q_2^2 + q_3^2} \tag{1}$$

where q_0, q_1, q_2, q_3 refer to the corresponding quaternions. All the hip joint angles belonging to related swimming strokes are normalized by Eq. (2) before being used as the input of

TABLE II: Label information and motion.

Motion	Encode
Backstroke	[1 0 0 0]
Breaststroke	[0 1 0 0]
Butterfly	[0 0 1 0]
Front Crawl	[0 0 0 1]

BiLSTM regression network

$$\begin{aligned} \theta_{lf\_\text{nor}} &= \frac{\theta_{lf-} - \theta_{If\_\text{mean}}}{\theta_{If\_\text{std}}} \\ \theta_{rf\_\text{nor}} &= \frac{\theta_{rf} - \theta_{rf\_\text{mean}}}{\theta_{rf\_\text{std}}} \end{aligned} \tag{2}$$

where θ_{lf} and θ_{rf} represent the raw data of hip angle, θ_{lf} nor and $\theta_{rf_}$nor indicate the normalized hip joint angles, θ_{mean} and θ_{std} mean the average and standard deviation of each leg of corresponding swimming stroke respectively.

Algorithm 1 The process of proposed control strategy

Load pre-trained DCNN-BiLSTM and four BiLSTM networks; Acquire sequential motion data of the swimmer M_t and A_t;

$motion = DCNN - BiLSTM(M_t)$;

if $motion = backstroke$:

$A_t\_estimated = BiLSTM\_backstroke(A_t)$;

else if $motion = backstroke$:

$A_t\_estimated = BiLSTM\_breaststroke(A_t)$;

else if $motion = butterfly$:

$A_t\_estimated = BiLSTM\_butterfly(A_t)$;

else if $motion = frontcrawl$:

$A_t\_estimated = BiLSTM\_frontcrawl(A_t)$;

end if

The pre-defined hip motion trajectory is:

$Trajectory = (A_1\_estimated, \ldots, A_t\_estimated)$

return $Trajectory$

III. Deep Learning Control Strategy

A. Control Strategy Overview

Fig. 4 presents the overview of proposed control strategy. Currently, the control strategy can be concluded as two steps:
Step 1: Performing swimming stroke recognition with acceleration and angular velocity of lumbar utilizing DCNN-BiLSTM classification network;
Step 2: According to the swimming stroke identified by Step 1, estimating the hip joint angles in advance using pitch calculated from two thigh IMUs by loading BiLSTM regression network that belongs to the identified motion.

From Fig. 4, after data preprocessing, the swimmer's current swimming stroke can be defined by the piecewise sequence of $m_t = \left(a_{1t}, \dot{\theta}_{1t}\right)$, where a_{lt} and $\dot{\theta}_{lt}$ refer to the acceleration and angular acceleration of the lumbar at time t. Sequential motion data for motion recognition M_t is combined of $M_t = (m_t, m_{t+1}, \ldots, m_{t+119})$. The normalized hip joint angle piecewise sequence can be used to define the swimmer's hip motion of current swimming stroke by $\theta_t = (\theta_{lh}, \theta_{rh})$, where θ_{lh} and θ_{rh} indicate the normalized hip joint angle of left and right hip at time t. Then sequential motion data for hip trajectory estimation A_t consists of $A_t = (\theta_t, \theta_{t+1}, \ldots, \theta_{t+119})$. Algorithm 1 below describes the process of proposed control strategy. Through Step 1 and 2 , a pre-defined hip motion trajectory of the recognized swimming stroke is generated to guide the lower limbs of the wearer through the power assistance facility.

B. Methods

CNN is usually adapted in DL network to conduct feature extraction on input data automatically. DCNN consists of several CNN layers. The input data are transferred to kernel with different size in a CNN layer. For example, a size of 3×1 CNN kernel shown in Fig. 5, which is optimized from the input data, mainly play a role of a supervised training process in order to maximize the activation level. Eq. (3) below describes the function of CNN kernels, where a_{ij} indicates the j th sample of layer 1 feature map, f refers to the Sigmoid function, $w_{m,n}$ means a $H \times K$ convolution kernel weight matrix, b illustrates the bias of layer 1 feature map, and $x_{i+m,j+n}$ shows the activation value of the upper neurons connected to the neuron (i, j).

$$a_{i,j} = f\left(\sum_{m=1}^{H}\sum_{n=1}^{K} W_{m,n} \cdot X_{i+m,j+n} + b\right) \tag{3}$$

A CNN layer often cooperates with a max-pooling layer for reducing feature dimensionality, preventing feature distortions and effectively identifying target labels. The output of max-pooling layer can be illustrated by Eq. (4) below, where R means the size of max-pooling layer and T is the corresponding pooling stride.

$$p_{i,j} = \max_{r \in R}\left(a_{i \times T+r,j}\right) \tag{4}$$

Some gated RNN methods such as GRU and LSTM are proposed to overcome the gradient vanishing or explosion of RNN dealing with long term study situations. GRU has been proved to have fewer parameters and shorter processing time. However, when handling large-scale datasets, LSTM achieves a better performance than GRU, which proceeds sequential information by memory cells that store and output information to learn long time scales temporal relationships [40]. BiLSTM, combined of bidirectional RNN and LSTM, considers information from both past and future, which can be seen in Fig. 6 below. Eq. (5) explains the information flow of a BiLSTM cell, where σ and g mean the Sigmoid function and Softmax function singly, $\tilde{c}^{[t]}, \Gamma_u, \Gamma_f, \Gamma_o, c^{[t]}$ represent the output of input modulation gate, update gate, forget gate, output gate and cell state respectively, $h^{[t]}$ refers to the output of single BiLSTM cell calculated from the old cell state $d^{[t]}$ and the future cell state $d^{[t+1]}$ independently, W_c, W_u, W_f and W_o are the corresponding coefficient matrix, b_c, b_u, b_f and b_o are the relevant bias vectors.

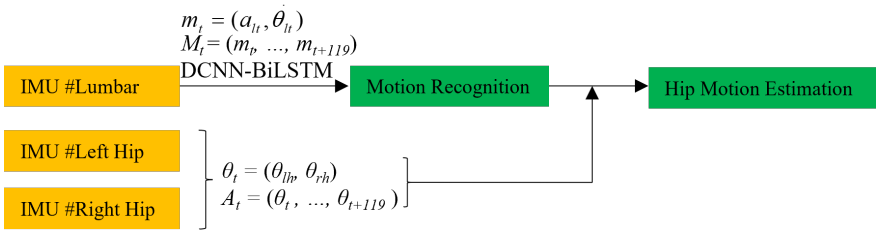


Fig. 4: The overview of control strategy.

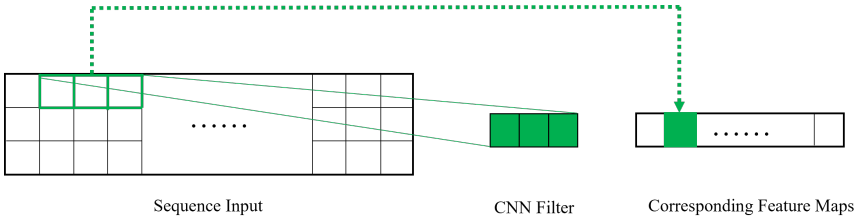


Fig. 5: The function of CNN filter.

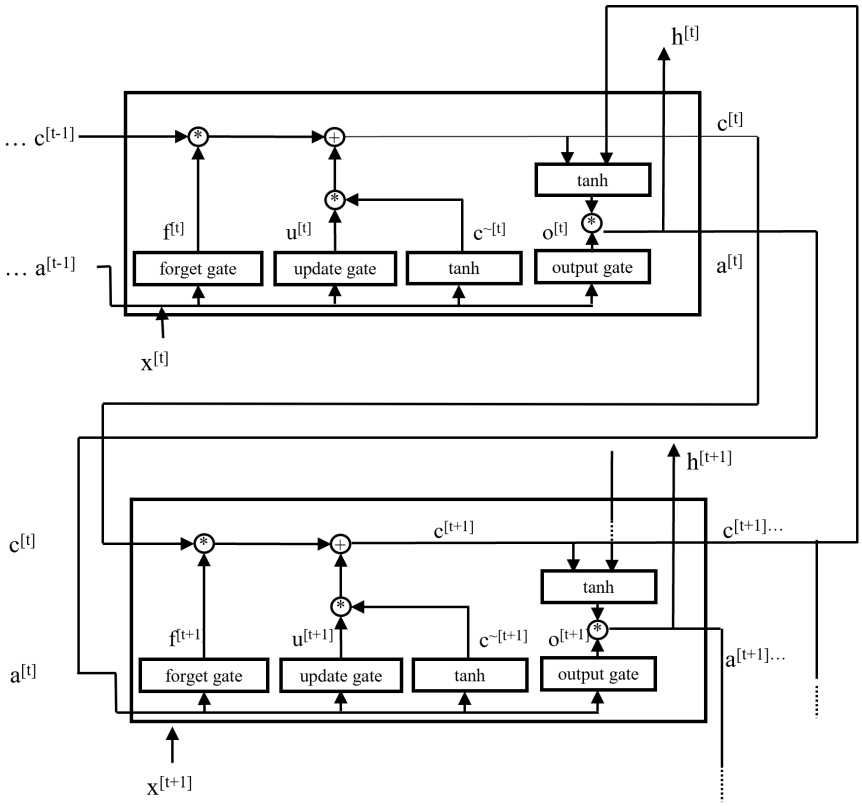


Fig. 6: The structure of BiLSTM. It can be seen that the output of each BiLSTM cell takes both the cell state $c^{[t]}$ and the future cell state $c^{[t+1]}$ into consideration.

$$\begin{cases} \tilde{c}^{[t]} = \tanh\left(W_c\left[a^{[t-1]}, x^{[t]}\right] + b_c\right) \\ \Gamma_u = \sigma\left(W_u\left[a^{[t-1]}, x^{[t]}\right] + b_u\right) \\ \Gamma_f = \sigma\left(W_f\left[a^{[t-1]}, x^{[t]}\right] + b_f\right) \\ \Gamma_o = \sigma\left(W_o\left[a^{[t-1]}, x^{[t]}\right] + b_o\right) \\ c^{[t]} = \Gamma_f \otimes c^{[t-1]} + \Gamma_u \otimes \tilde{c}^{[t]} \\ H^{[t]} = \Gamma_o \otimes \tanh\left(c^{[t]}, c^{[t+1]}\right) \end{cases} \tag{5}$$

C. Motion Recognition

Fig. 7 illustrates the structure of DCNN-BiLSTM classification network, which combines of two groups of CNN and max-pooling layer, two BiLSTM layers and two full connection layers (FC). The sequence input of the network has a size of 6×120. The kernel size of convolutional layer is 1×, and the size of max-pooling layer is 1×2 with a stride of 1×2. The number of hidden neurons of each BiLSTM layer is

16. A dropout layer exists between two FC layers and the possibility is set to 0.4 in order to prevent overfitting. We choose SGD optimizer with initial learning rate of 0.01 as the gradient descent method. The max epoch value is set to 50 during training. The activation function of DCNN-BiLSTM is Softmax. For more details like the cost function, the adjustment of hyper-parameters etc. of DCNN-BiLSTM, please refer to our previous work [14]. The performance of the classification network can be evaluated by Eq. (6), where TP, FP, TN and FN refer to true positive, false positive, true negative and false negative independently.

$$\begin{aligned} \text{accuracy} &= \frac{TP+FN}{TN+TP+FP+FN} \\ \text{precision} &= \frac{TP}{TP+FP} \\ \text{recall} &= \frac{TP}{TP+FN} \end{aligned} \tag{6}$$

To explain the forward propagation of proposed DCNN-BiLSTM, firstly, the feature matrix $\boldsymbol{p^l}$ from CNN and max-pooling layer is calculated according to Eq. (3) and Eq. (4). Then, the output of BiLSTM layer $h^l = \left[h^{[1]}, \ldots, h^{[l]}\right]$ is generated from p^l according to Eq. (5). After that, the full connected (FC) layers proceed the BiLSTM output by Eq. (7) below, where w_{ji}^{l-1} and b_i^{l-1} are the weight matrix and bias term between the ith node of layer $l-1$ and the j th node of layer l respectively. Finally, the output of classification layer, which is calculated by Softmax function, generates the classification results according to the output of FC layers f^l and Eq. (8)

$$f^l = \sum_j w_{ji}^{l-1}\left(\sigma\left(h_i^{[l-1]}\right) + b_i^{l-1}\right) \tag{7}$$

$$\hat{y}_j^{(i)} = \arg\max_{j\in i} \frac{1}{1+\exp\left(-W^{(l)T} f^l\right)} \tag{8}$$

where $\hat{y}_j^{(i)}$ means the predicted possibility for the sample j belonging to class i. The cost function J of DCNN-BiLSTM is illustrated in Eq. (9)

$$J = \frac{1}{M}\sum_{i=1}^{M} L\left(\hat{y}_j^{(i)}, y_j^{(i)}\right) \tag{9}$$

where M refers to minibatch size, $y_j^{(i)}$ indicates the true possibility for the sample j belonging to class i, L is cross entropy lost function in the form of Eq. (10).

$$L = -\frac{1}{M}\sum_{j=1}^{M}\sum_{i=1}^{N} y_j^{(i)} \log \hat{y}_j^{(i)} \tag{10}$$

D. Motion Estimation

Fig. 8 illustrates the architecture of BiLSTM regression network in this work, which is made up of two BiLSTM layers, two FC layers, a dropout layer and a regression layer. The sequence layer size here becomes 2×120. Each BiLSTM layer has 8 hidden neurons. The gradient descent with Adam optimizer is chosen with an initial learning rate of 0.01, and the max epoch is 40 during training. The dropout possibility of the dropout layer between two FC layers is set to 0.2 through multiple attempts. The weight decay which is used for L2-Regularization to prevent network overfitting [41] of backstroke, breaststroke, butterfly and front crawl network are adapted to 0.014, 0.034, 0.01 and 0.026 respectively according to the results of several trails. The performance of BiLSTM conducting hip joint angle estimation can be evaluated by root mean square error (RMSE) and association coefficient (R) between the actual and estimated values, just as expressed by Eq. (11)

$$\begin{aligned} RMSE &= \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - y_i)^2} \\ R &= 1 - \frac{\sum_{i=1}^{N}(x_i - y_i)^2}{\sum_{i=1}^{N}\left(x_i - \widehat{X_i}\right)^2} \end{aligned} \tag{11}$$

where N refers to the number of input sequence data, x_i, y_i indicate the unnormalized input and predicted values respectively.

The forward propagation of proposed BiLSTM regression network can be explained as follows. The output of BiLSTM layer is calculated from sequence input by Eq. (5) and transferred to full connected (FC) layers by Eq. (7). The output of regression layer, which is calculated by hyperbolic tangent function (tanh), generates the estimated hip joint angles. The mean squared error function (MSELoss) described in Eq. (12) is chosen as the lost function.

$$L = \{I_1, \ldots, I_n\}, l_i = (x_i - y_i)^2 \tag{12}$$

IV. Results

A. Offline Testing

As mentioned in Section II.A that each subject generates four groups of data of four swimming strokes. We randomly select one group data of each motion as the test set and the remaining data as the training set during offline training and testing. The major concern of offline testing is the accuracy of motion recognition and estimation of DCNN-BiLSTM classification network and four BiLSTM regression networks. The motion recognition and estimation are conducted separately during the offline testing.

1) Motion Recognition: To evaluate the performance of proposed DCNN-BiLSTM, some hybrid DL networks such as ELM and CNN-GRU used by Vijay et al. [15], [17] are also utilized here for comparison experiments. Fig. 9(a)-(c) present the confusion matrixes of test set utilizing DCNN-BiLSTM, CNN-GRU and ELM, achieving overall recognition accuracy of 96.53%, 94.49% and 93.77% respectively. From the confusion matrixes of three DL methods in Fig. 9 that some breaststroke motions are tend to be misidentified to butterfly, and this may be caused by the similarity lumbar motion of breaststroke and butterfly [42]. The recognition report of three DL method can be seen in Tab. III (a)-(c) below. From Tab. III , three chosen DL method all achieve a satisfying result of backstroke and front crawl recognition. The backstroke recognition receives an accuracy of more than 98.99%, a precision of more than 97% and a recall of

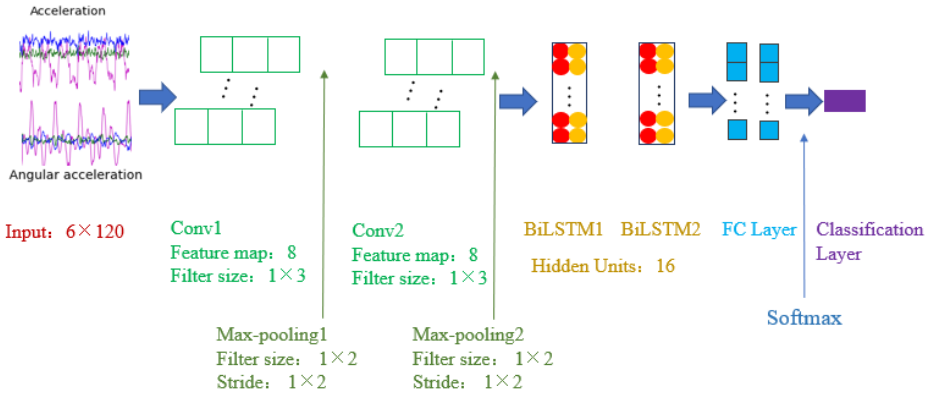


Fig. 7: The structure of DCNN-BiLSTM classification network.

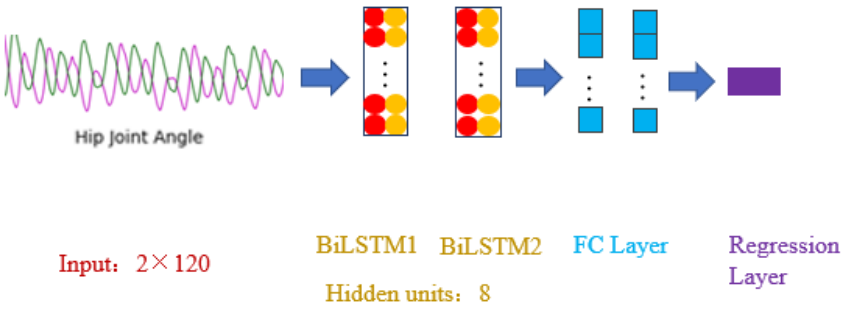


Fig. 8: The structure of BiLSTM regression network.

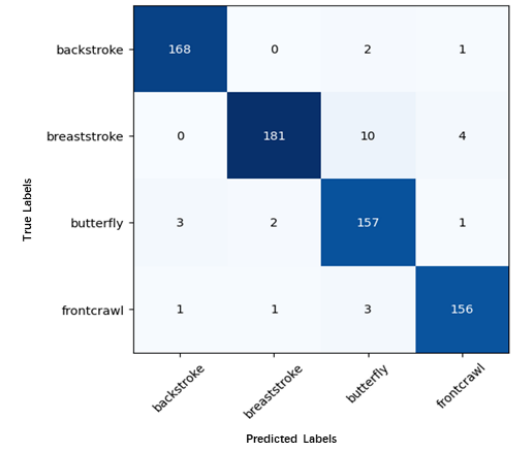
more than 98.25%, while the corresponding results of front crawl recognition are more than 97.68%, 94.44% and 92.44% respectively. However, our proposed DCNN-BiLSTM greatly outperforms during breaststroke and butterfly recognition. As can be seen in Tab. III , except for the recall of butterfly, DCNN-BiLSTM obtains the best results among three DL methods. Considering the weakness of ELM and CNN-GRU, conclusion can be drawn that the proposed DCNN-BiLSTM is capable of conducting swimming stroke recognition in our control strategy.

2) Motion Estimation: Fig. 10(a)-(d) present the actual and estimated hip joint angles and errors between them of backstroke, breaststroke, butterfly and front crawl of three subjects respectively utilizing proposed BiLSTM regression network. Data that unrelated to the recognized motion (i.e. the swimmer getting into the swimming pool from the bank) at the beginning and the end of test set are not presented in Fig. 10. From the curves shown in Fig. 10, we discover that the variation tendency and values of actual hip joint angles can keep the pace with those of the estimated hip joint angles. However, it is obviously seen that the errors between actual and estimated angles of butterfly shown in Fig. 10(c) are larger compared with other three swimming strokes. This may because that leg motions of each butterfly cycle are more complex to conduct that during one cycle, the swimmer needs to finish two leg kicks, which raises the motion requirement for swimmers and restricts the quality of collected data.

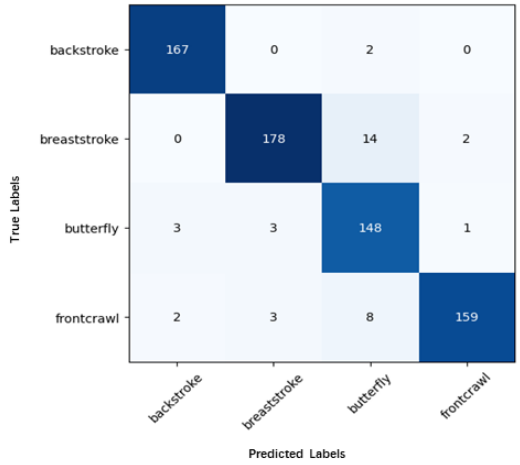
A GRU regression network is also adapted here for comparison experiment. The motion estimation performance utilizing BiLSTM and GRU are listed in Tab. IV (a)-(b) below independently. The overall RMSE of BiLSTM and GRU are 0.9297° and 1.0148°, and BiLSTM outperforms about 0.09° in RMSE than GRU. It is seen from Tab. IV that BiLSTM achieves a better result than GRU among most of the items in motion estimation report. The relatively worse RMSE and R of butterfly shown in Tab. IV below validate the conclusion we draw from Fig. 10 again. Generally speaking, we have achieved satisfying RMSE less than 1.2° and R more than 0.93, which prove that the proposed BiLSTM regression network of four swimming strokes can predict the hip motion trajectory well.

B. Online Testing

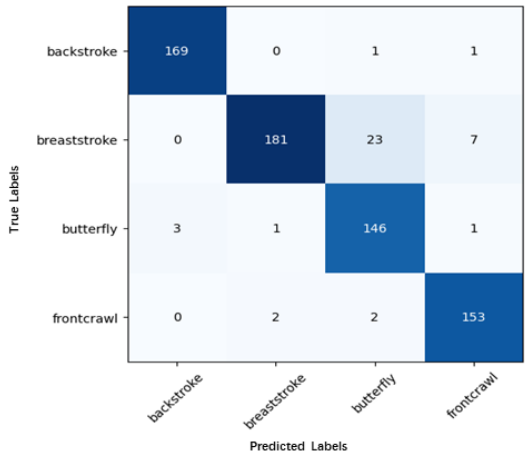
1) Online Testing Setup: In online testing, the pre-trained DCNN-BiLSTM classification network and four BiLSTM regression networks are transplanted into a Raspberry Pi 4B according to Algorithm 1 to realize our control strategy. Because we have verified the reliability of proposed DL methods, considering the complexity of conducting online testing, we do not design comparison experiments here. We invite subject 2 again to participate our online testing experiment and swim in the order of backstroke, breaststroke, butterfly and front crawl for 50m independently. A battery, a lumbar IMU and Raspberry Pi 4B are placed in a waterproof box on the



(a) DCNN-BiLSTM



(b) CNN-GRU



(c) ELM

Fig. 9: The confusion matrix of swimming stroke recognition utilizing (a). DCNN-BiLSTM; (b). CNN-GRU and (c). ELM.

swimmer's waist, while other two IMUs are placed on the swimmer's thighs.

Both the accuracy and robustness are the key point in online testing. The sliding window length and overlap of four

TABLE III

(a) The motion recognition report utilizing DCNN-BiLSTM.

Motion	Backstroke	Breaststroke	Butterfly	Front crawl
Accuracy	0.9899	0.9754	0.9697	0.9841
Precision	0.9767	0.9837	0.9128	0.9630
Recall	0.9825	0.9282	0.9632	0.9689

(b) The motion recognition report utilizing CNN-GRU.

Motion	Backstroke	Breaststroke	Butterfly	Front crawl
Accuracy	0.9899	0.9681	0.9565	0.9768
Precision	0.9709	0.9674	0.8604	0.9815
Recall	0.9881	0.9175	0.9668	0.9244

(c) The motion recognition report utilizing ELM.

Motion	Backstroke	Breaststroke	Butterfly	Front crawl
Accuracy	0.9928	0.9522	0.9551	0.9812
Precision	0.9826	0.9837	0.8488	0.9444
Recall	0.9883	0.8578	0.9669	0.9745

TABLE IV

(a) The motion estimation report utilizing BiLSTM.

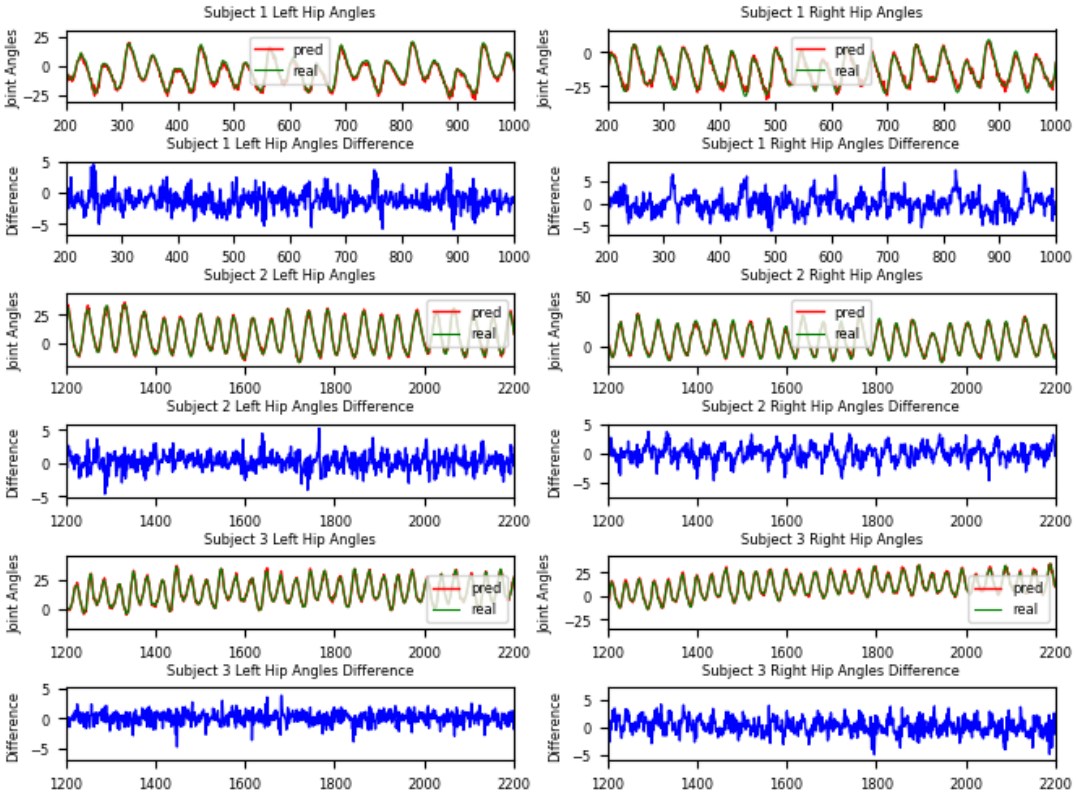
Motion / Subject	Backstroke		Breaststroke		Butterfly		Front crawl	
	RMSE	R	RMSE	R	RMSE	R	RMSE	R
1	0.7743°	0.9646	0.9012°	0.9610	1.0791°	0.9484	0.8943°	0.9522
2	0.9864°	0.9307	0.8561°	0.9728	1.1245°	0.9629	0.8612°	0.9706
3	0.8721°	0.9511	0.8662°	0.9688	0.9934°	0.9660	0.8759°	0.9723

(b) The motion estimation report utilizing GRU.

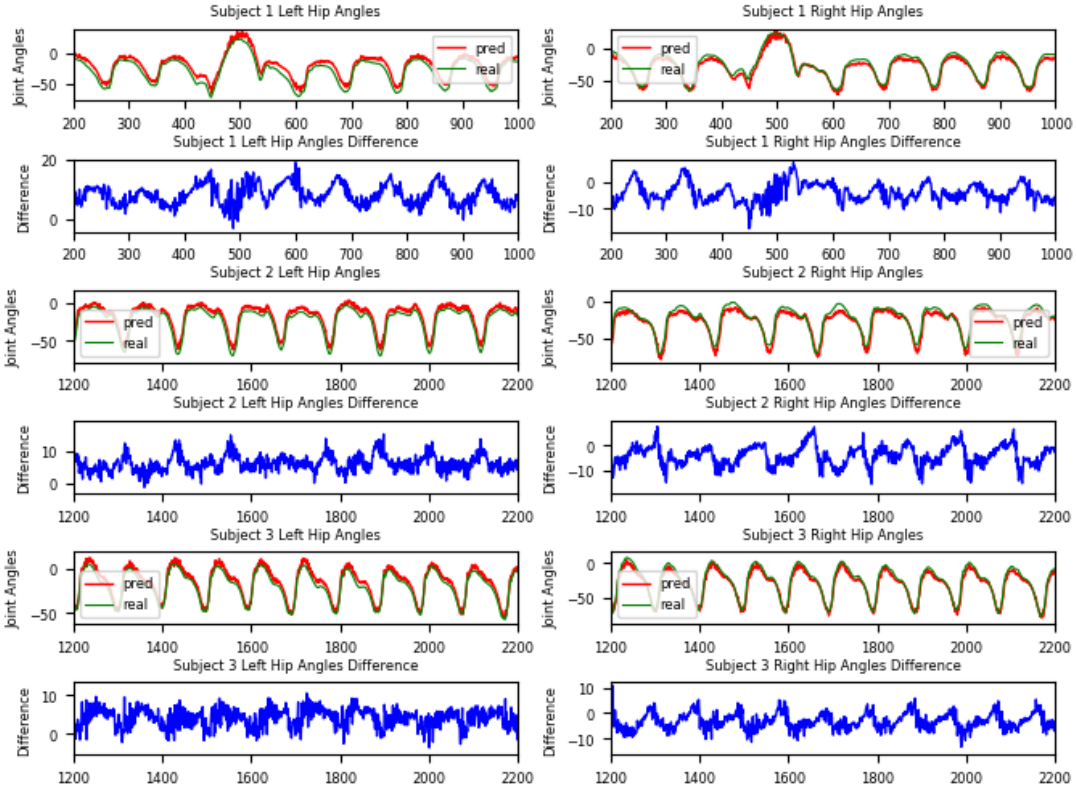
Subject / Motion	Backstroke		Breaststroke		Butterfly		Front crawl	
	RMSE	R	RMSE	R	RMSE	R	RMSE	R
1	0.8943°	0.9553	0.9827°	0.9489	1.1032°	0.9317	0.8543°	0.9627
2	1.0129°	0.9281	1.2188°	0.9412	1.1178°	0.9697	0.9986°	0.9510
3	0.8688°	0.9536	0.9739°	0.9501	1.1008°	0.9575	0.8934°	0.9639

BiLSTM becomes 120 and 110 timestamps in order to have a more detailed estimation of the swimmer's hip motion, while those of DCNN-BiLSTM are not changed. According to the sampling frequency (120Hz) of IMU, one classification result and six corresponding regression results are generated within one second during online testing.

2) Online Testing Results: Fig. 11(a)-(c) below present the related results of online testing experiments. Through Fig. 11(a), we discover that the overall online swimming stroke recognition accuracy is good. However, the recognition accuracy results of breaststroke and butterfly are needed to be improved. Besides, we find that identification errors tend to take place in the data period belonging to motion transitions like the end of butterfly and the beginning of front crawl. Because the swimmer is asked to continuously swim in one swimming stroke for 50m, the motions of this time period may be turning back or gliding for the preparation of the next swimming stroke, which are not recorded and labeled during

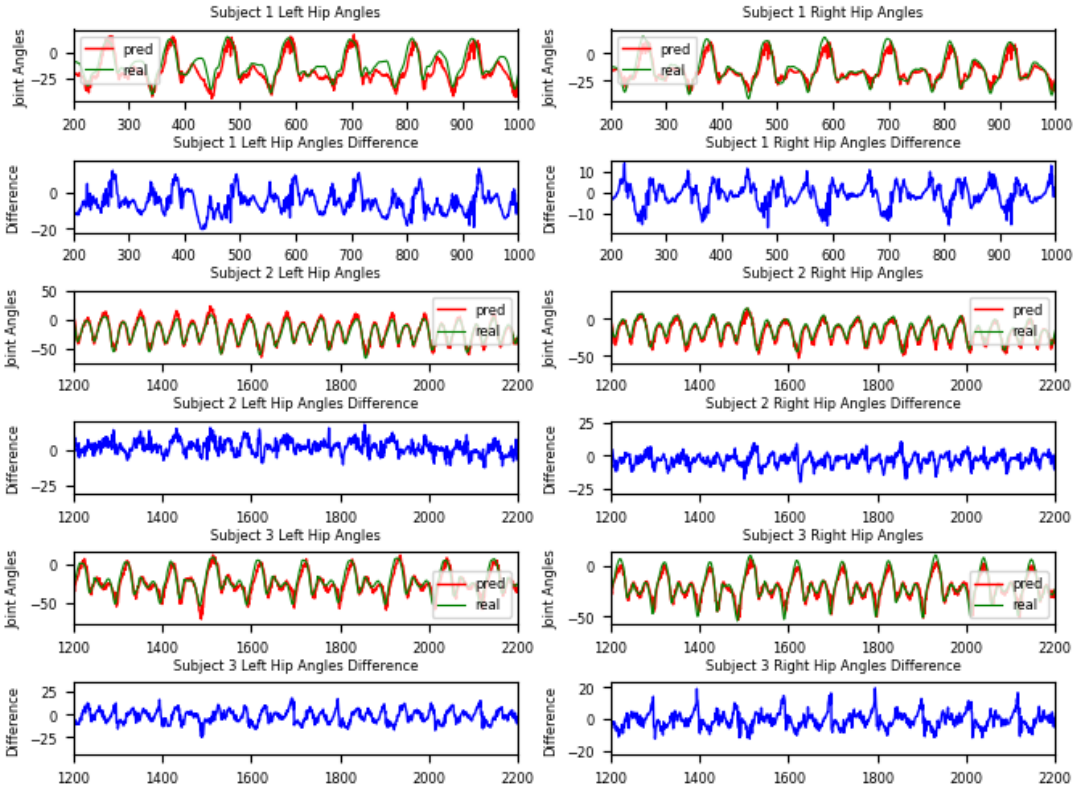


(a) Backstroke hip joint angle estimation.

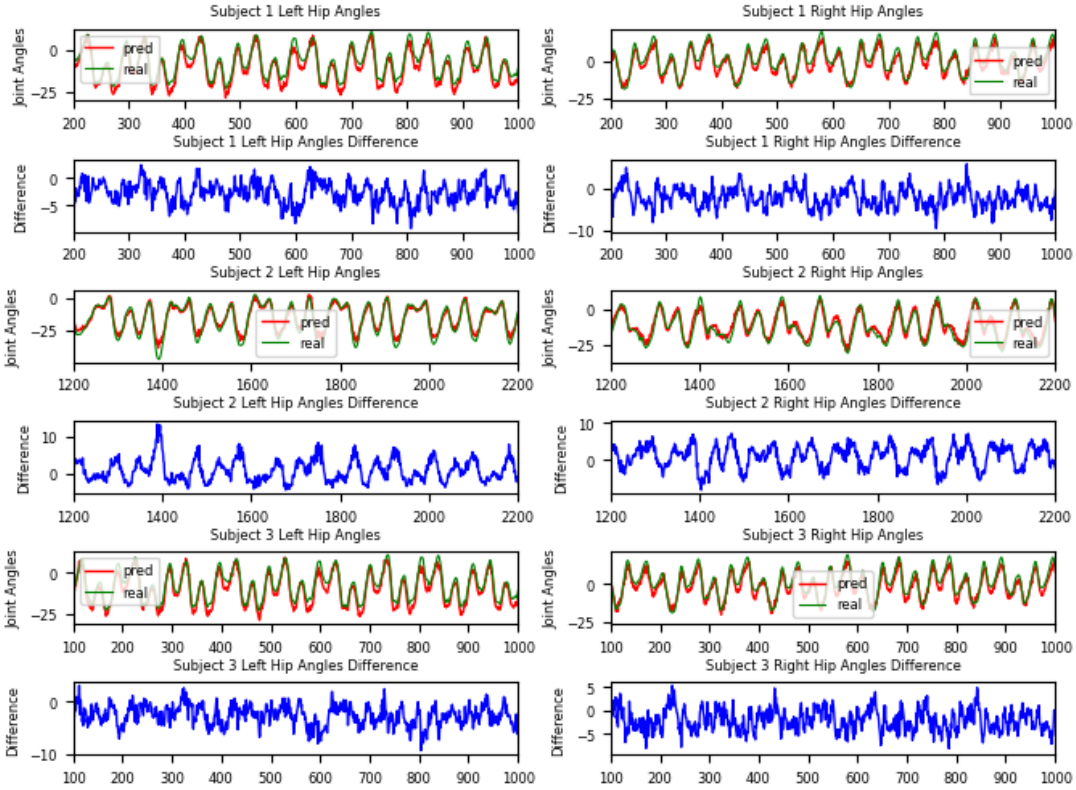


(b) Breaststroke hip joint angle estimation.

Fig. 10: The actual and estimated hip joint angles and errors between them of (a). backstroke, (b). breaststroke, (c). butterfly and (d). front crawl of three subjects, where X labels indicate the timestamps and Y labels indicate the values of predicted and actual hip joint angles and the difference between them.



(c) Butterfly hip joint angle estimation.



(d) Front crawl hip joint angle estimation.

Fig. 10: The actual and estimated hip joint angles and errors between them of (a). backstroke, (b). breaststroke, (c). butterfly and (d). front crawl of three subjects, where X labels indicate the timestamps and Y labels indicate the values of predicted and actual hip joint angles and the difference between them.

data acquisition and offline testing.

From Fig. 11(b)-(c), it can be seen that the performance of online motion estimation basically keeps the level of that in offline testing. In Fig. 11(b), we can find that the errors of left hip during butterfly is larger than those during other swimming strokes. The causes can be various, such as the wrong identification of butterfly motions just as illustrated at about 580-820 timestamps in Fig. 11(a), the IMU location deviation due to the water flow, or the nonstandard left leg butterfly kicks of the swimmer etc. However, it seems that the occasional misidentifications of other three swimming strokes do not have too much bad influence on the following hip motion estimation according to the errors shown in Fig. 11(b)-(c). Generally speaking, the online hip motion estimation also achieves an acceptable performance.

It is known that the user will produce delayed sensations if the time interval between the action of exoskeleton and human is more than 300ms [43]. For the robustness of DCNN-BiLSTM and BiLSTM, it takes average 8.47ms for DCNN-BiLSTM to take a motion recognition, while average 6.72ms is needed for BiLSTM to proceed an angle estimation, which are much less than the length of sliding window (120ms) and required time interval.

V. Discussion and Conclusion

In this study, considering the mode of lower limb generating propelling force in four swimming strokes, we propose a DL control strategy based on our previous work aiming at hip power-assisted swimming exoskeleton utilizing IMUs. According to the biggest challenge of exoskeleton control, our control strategy greatly overcomes the limitations of phase detected control strategy, and make contributions to the underwater human motion and underwater power-assisted wearable exoskeleton research.

It has been proved through our experiment results that the control strategy can recognize and estimate the wearer's motion with satisfying accuracy and robustness, and can cope with the ruleless motion switches of the wearer. The current problem is also clear that the algorithm performance is needed to be improved when implemented on butterfly mode. As we mentioned before that, the mechanism of lumbar motion of breaststroke and butterfly is similar, which contributes to the not that satisfying recognition accuracy of butterfly. Meanwhile, the leg motion of one butterfly cycle are more complex for swimmers to conduct, and recognizing butterfly as breaststroke lead to the relatively large error between estimated and real hip joint angles. Generally speaking, the advantages of proposed control strategy are concluded as follows:

(1) When applied to exoskeletons, the proposed control strategy is able to correctly recognize and estimate the wearer's motion with a quick response, so that the user will not feel a time delay of the action between him/her and the exoskeleton;

(2) The hardware of proposed control strategy is only combined of a battery, a Raspberry Pi 4B and three IMUs, making it low-cost, low-weight, high-sensitivity and easy to be applied;

(3) The proposed control strategy can be used to exoskeleton in any field such as rehabilitation exoskeletons as long as acquiring a dataset of target motions (i.e. easy for usage transferring);

(4) The random and discontinuous errors of swimming stroke recognition do not show obvious effect on the hip joint angle estimation of the swimmer (i.e. strong fault tolerance of control strategy).

However, according to the experiment results, our control strategy still has the following limitations:

(1) The butterfly recognition performance of DCNN-BiLSTM are needed to be optimized. Another IMU that placed at the back, or the head of the swimmer is needed to generate more data feature for motion recognition performance improvement. Also, the number of subject participating our data collection experiment is too small. Data of swimmers of different ages, genders and swimming levels are needed to enlarge the features of our dataset;

(2) Just as mentioned in Section IV.B.2 that, the transition data between different motions deserve a label during data preprocessing for further improvement of motion recognition accuracy. This is a fatal drawback of our proposed control strategy, because the swimmer cannot swim all the time. The control strategy should recognize the rest, and preparation motions of the swimmer such as turn around and gliding in order not to conduct hip motion estimation. Otherwise, when applied to power-assisted exoskeletons, providing power assistance during transition motion periods will make the wearer injured;

(3) The environment of a swimming pool is mild. It remains to be seen that how the control strategy performs in open water when applied to exoskeletons, which is a dangerous experiment and hard for us to conduct currently.

In fact, based on the two steps shown in this study, our control strategy still has a third step to be realized in the future:

Step 3: According to the estimated hip joint angles of Step 2, send the pre-defined motion trajectory to the motor for assistance control.

We are now devoted to working on the research and development of a Bowden cable driven hip power-assisted swimming exoskeleton. Currently, our work is still in the stage of related hardware selection (e.g. motor type etc.) and exoskeleton structure CAD design. When the prototype development is finished, the proposed control strategy overcoming current limitations will definitely be embedded to it. With the help of our exoskeleton, we believe swimmers can swim easier and longer.

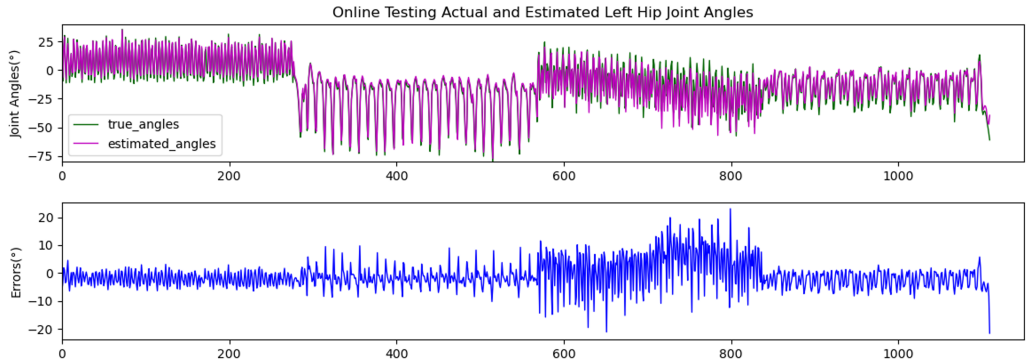
Acknowledgment

The author would like to thank the Hunan Youth Swimming Team for providing experimental site and three elite swimmers for this study.

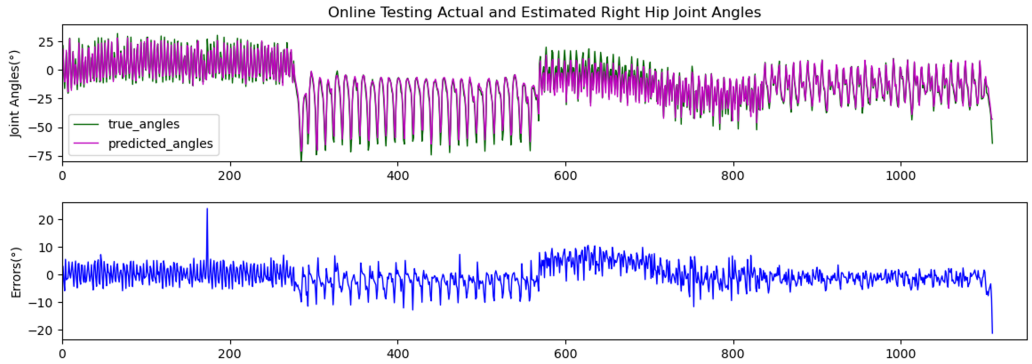
Declaration of Interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

(a) Online motion recognition results.



(b) Online left hip joint angle results.



(c) Online right hip joint angle results.

Fig. 11: The online testing results of (a). motion recognition, where label of 1, 2, 3, 4 represent backstroke, breaststroke, butterfly and front crawl respectively for recording convenience; (b). the actual and estimated left hip joint angles and the errors between them; (c). the actual and estimated right hip joint angles and the errors between them

REFERENCES

[1] Low, K., X. Liu, and H. Yu. *Development of NTU wearable exoskeleton system for assistive technologies. in IEEE International Conference Mechatronics and Automation, 2005.* 2005. IEEE.

[2] Chen, B., et al.,*A wearable exoskeleton suit for motion assistance to paralysed patients.* Journal of orthopaedic translation, 2017. 11: p. 7-18.

[3] Hsu, S.-W. and T.-T. Lin.*Development and assessment of a wearable waist-assistive exoskeleton for agricultural tasks. in 2021 ASABE Annual International Virtual Meeting.* , 2021. American Society of Agricultural and Biological Engineers.

[4] Jia-Yong, Z., et al. *A preliminary study of the military applications and future of individual exoskeletons. in Journal of Physics: Conference Series.* ., 2020. IOP Publishing.

[5] Foroutannia, A., M.-R. Akbarzadeh-T, and A. Akbarzadeh, *A deep learning strategy for EMG-based joint position prediction in hip exoskeleton assistive robots.* Biomedical Signal Processing and Control, 2022. 75: p. 103557.

[6] Rebutini, V.Z., et al., *Plyometric long jump training with progressive loading improves kinetic and kinematic swimming start parameters.* Journal of Strength and Conditioning Research, 2016. 30(9): p. 2392-2398.

[7] Chen, L., D. Hu, and X. Han,*Study on forearm swing recognition algorithms to drive the underwater power-assisted device of frogman.* Journal of Field Robotics, 2022. 39(1): p. 14-27.

[8] Wang, Q., et al.,*An underwater lower-extremity soft exoskeleton for breaststroke assistance.* IEEE Transactions on Medical Robotics and Bionics, 2020. 2(3): p. 447-462.

[9] Yu, S., et al., *Design and control of a high-torque and highly backdrivable hybrid soft exoskeleton for knee injury prevention during squatting.* , IEEE Robotics and Automation Letters, 2019. 4(4): p. 4579-4586.

[10] Di Natali, C., et al.,*Design and evaluation of a soft assistive lower limb exoskeleton.* Robotica, 2019. 37(12): p. 2014-2034.

[11] Hernandez, V., et al., *Lower body kinematics estimation from wearable sensors for walking and running: A deep learning approach.* , Gait & Posture, 2021. 83: p. 185-193.

[12] Argent, R., et al., *Evaluating the use of machine learning in the assessment of joint angle using a single inertial sensor.* Journal of Rehabilitation and Assistive Technologies Engineering, 2019. 6: p. 2055668319868544.

[13] Zhao, R., et al., *Machine health monitoring using local feature-based gated recurrent unit networks.* IEEE Transactions on Industrial Electronics, 2017. 65(2): p. 1539-1548.

[14] Chen, L. and D. Hu, *An Effective Swimming Stroke Recognition System Utilizing Deep Learning Based on Inertial Measurement Units (accepted but not published).* Advanced Robotics, 2022.

[15] Dua, N., et al., *Inception inspired CNN-GRU hybrid network for human activity recognition.* Multimedia Tools and Applications, 2022: p. 1-35.

[16] Semwal, V.B., et al., *An optimized feature selection using bio-geography optimization technique for human walking activities recognition.* Computing, 2021. 103(12): p. 2893-2914.

[17] Patil, P., et al. *Clinical human gait classification: extreme learning machine approach. in 2019 1st international conference on advances in science, engineering and robotics technology (ICASERT).* 2019. IEEE.

[18] Semwal, V.B., et al., *Pattern identification of different human joints for different human walking styles using inertial measurement unit (IMU) sensor.* Artificial Intelligence Review, 2022. 55(2): p. 1149-1169.

[19] Challa, S.K., et al.,*An Optimized-LSTM and RGB-D Sensor-Based Human Gait Trajectory Generator for Bipedal Robot Walking.* IEEE Sensors Journal, 2022. 22(24): p. 24352-24363.

[20] Raj, M., V.B. Semwal, and G.C. Nandi, *Bidirectional association of joint angle trajectories for humanoid locomotion: the restricted Boltzmann machine approach.* Neural Computing and Applications, 2018. 30: p. 1747-1755.

[21] Jain, R., V.B. Semwal, and P. Kaushik, *Stride segmentation of inertial sensor data using statistical methods for different walking activities. Robotica,* 2022. 40(8): p. 2567-2580.

[22] Bijalwan, V., et al., *HDL-PSR: Modelling spatio-temporal features using hybrid deep learning approach for post-stroke rehabilitation.* Neural Processing Letters, 2022: p. 1-20.

[23] Jain, R. and V.B. Semwal, *A Novel Feature Extraction Method for Preimpact Fall Detection System Using Deep Learning and Wearable Sensors.* IEEE Sensors Journal, 2022. 22(23): p. 22943-22951.

[24] Chen, Y.-L., et al., *IMU-based Estimation of Lower Limb Motion Trajectory with Graph Convolution Network.* IEEE Sensors Journal, 2021. 21(21): p. 24549-24557.

[25] Chen, S., et al. *Real-time walking gait estimation for construction workers using a single wearable inertial measurement unit (imu). in 2021 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM).* 2021. IEEE.

[26] Sung, J., et al., *Prediction of Lower Extremity Multi-Joint Angles during Overground Walking by Using a Single IMU with a Low Frequency Based on an LSTM Recurrent Neural Network.* Sensors, 2021. 22(1): p. 53.

[27] Ren, J.-L., et al. *Deep learning based motion prediction for exoskeleton robot control in upper limb rehabilitation. in 2019 International Conference on Robotics and Automation (ICRA).* 2019. IEEE.

[28] Zhu, L., et al., *A novel motion intention recognition approach for soft exoskeleton via IMU.* Electronics, 2020. 9(12): p. 2176.

[29] Nomura, S., et al., *Power assist control based on human motion estimation using motion sensors for powered exoskeleton without binding legs.* Applied Sciences, 2019. 9(1): p. 164.

[30] Wang, J., et al., *Integral real-time locomotion mode recognition based on GA-CNN for lower limb exoskeleton.* Journal of Bionic Engineering, 2022. 19(5): p. 1359-1373.

[31] Long, Y., et al., *Human motion intent learning based motion assistance control for a wearable exoskeleton.* Robotics and Computer-Integrated Manufacturing, 2018. 49: p. 317-327.

[32] Fang, B., et al., *Gait neural network for human-exoskeleton interaction.* Frontiers in Neurorobotics, 2020. 14: p. 58.

[33] Lee, D., et al.,*Real-time user-independent slope prediction using deep learning for modulation of robotic knee exoskeleton assistance.* IEEE Robotics and Automation Letters, 2021. 6(2): p. 3995-4000.

[34] Yoo, M., et al., *Motion estimation and hand gesture recognition-based human–UAV interaction approach in real time.* Sensors, 2022. 22(7): p. 2513.

[35] Luo, K., et al., *A dual-scale morphological filtering method for composite damage identification using FBP.* Mechanical Systems and Signal Processing, 2023. 184: p. 109683.

[36] Wu, G., et al., *ISB recommendation on definitions of joint coordinate system of various joints for the reporting of human joint motion—part I: ankle, hip, and spine.* Journal of biomechanics, 2002. 35(4): p. 543-548.

[37] Wu, G., et al., *ISB recommendation on definitions of joint coordinate systems of various joints for the reporting of human joint motion—Part II: shoulder, elbow, wrist and hand.* Journal of biomechanics, 2005. 38(5): p. 981-992.

[38] Kautz, T., et al., *Activity recognition in beach volleyball using a deep convolutional neural network.* Data Mining and Knowledge Discovery, 2017. 31(6): p. 1678-1705.

[39] Xu, C., et al., *InnoHAR: A deep neural network for complex human activity recognition.* Ieee Access, 2019. 7: p. 9893-9902.

[40] Ordóñez, F.J. and D.J.S. Roggen, *Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition.* 2016. 16(1): p. 115.

[41] Gupta, S., et al. *A comparative analysis of various regularization techniques to solve overfitting problem in artificial neural network. in International conference on recent developments in science, engineering and technology.* 2017. Springer.

[42] McCabe, C., B. Mason, and J. Fowlie. *A temporal investigation into the butterfly kick placement following a breaststroke start and turn. in ISBS-Conference Proceedings Archive.* 2012.

[43] Englehart, K. and B. Hudgins, *A robust, real-time control scheme for multifunction myoelectric control.* IEEE transactions on biomedical engineering, 2003. 50(7): p. 848-854.