Evaluating the Effectiveness of Physical Rehabilitation Exercises through RehabNet++ and Hybrid Optimization Techniques

Osamah A. Altammami[1,*]

[1]College of Business Administration, Majmaah University, Majmaah 11952, Saudi Arabia

Email: o.altammami@mu.edu.sa

Abstract

This article focuses on improving the accuracy and efficiency of multimodal human motion analysis using advanced techniques. Initially, Generative Adversarial Networks (GANs) were used for skeletal enhancement, and then Contrast-Limited Adaptive Histogram Equalization (CLAHE) was applied on the enhanced images to check the quality Joint-level. Limb-level, Temporal, Statistical Features are effectively recovered from contrast enhancing images. Furthermore, with the selected optimal features acquired from PutterFish Customized Serval Optimizer (PFCSO), the RehabNet++ architecture that makes the human movement assessment has been trained. This PFCSO model has been developed based on the inspiration acquired from the Pufferfish Optimization Algorithm (POA) and the Serval Optimization algorithm (SOA), respectively. The RehabNet++ architecture includes an optimized Multilayer Perceptron (O-MLP), STR-ResNet architecture, Attention-based Convolutional Neural Networks and Transfer Learning. The O-MLP model has been formulated by optimizing the hidden layers of MLP using the PFCSO model. In addition, Grad-CAM visualization is included to provide a graphical description for model selection. A comparative study has been conducted to test the proposed deep learning algorithm against the original methods using the Kimore dataset. This analysis is implemented in PYTHON and is dedicated to multimodal human motion analysis.

Keywords: Physical Rehabilitation; Generative Adversarial Network; PFCSO; O-MLP; RehabNet++

1. Introduction

The significant impact of the current epidemic on mental and physical health, and the importance of exercise and physical activity, cannot be emphasized enough. During these critical times, it is important for people to emphasize physical activity, as these activities are crucial to mitigate the effects of the epidemic [1]. Task-specific physiotherapy exercises are the best way to diagnose musculoskeletal disorders and help stroke victims recover them [2]. Recent research illustrates the fact that fatigue is very common among individuals who experience stroke, even in cases of complete recovery [3]. Hence, it is vitally important to develop and evaluate methods to facilitate physical rehabilitation. Research shows that developing novel methods for physical rehabilitation can result in higher efficacy and longer commitment [4]. Furthermore, in a systematic review study to compare virtual/augmented reality versus traditional methods for teaching physiotherapy, [5] found that the use of modern technology can stimulate learning and enhance adherence. Customized rehabilitation activities have been shown to improve older people's ability to perform physical activity and daily living [6]. Physical activity can have a positive effect on many cardiovascular risk factors, including weight, blood pressure, insulin sensitivity, and more [7]. Post-operative rehabilitation, especially following joint disease surgery, is a critical phase in the recovery process. The development of a precise and dependable rehabilitation model is essential. It helps alleviate strain on healthcare professionals, and enhances the appeal of rehabilitation by promoting adherence, monitoring, accessibility, and long-term sustainability [8] [9].

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

As individual’s age or encounter injuries and cognitive decline, postural imbalance and gait abnormalities are common occurrences. Early diagnosis and swift rehabilitation are paramount challenges. Utilizing human walking activities can aid in rectifying walking deformities and preserving postural stability. During exercise rehabilitation, real-time patient assessment by rehabilitation physicians ensures the training aligns with the patient's recovery progress. This dynamic approach enhances rehabilitation efficacy [10]. Pulmonary rehabilitation (PR) is instrumental for individuals with chronic respiratory conditions, promoting optimal physical and psychosocial health, fostering health-focused behaviors, and supporting disease management. Current research in neurological rehabilitation highlights the central nervous system's remarkable adaptability, offering hope for restoring limb function impaired by moderate stroke through scientific rehabilitation training [11] [12].

A customized, progressive rehabilitation intervention targeting various physical function domains notably enhanced the physical well-being of elderly hospitalized patients with acute decompensated heart failure compared to standard care [13]. Enhancing physical fitness emerges as a critical approach to counteract the influence of an unhealthy lifestyle on the mental well-being of adolescents and young adults. Fostering recreational and competitive sports engagement is pivotal in averting mental disruptions during this crucial life phase. Robust ensemble classification and deep learning for action recognition enhance precision and create comprehensive models [14] [15]. Rehab-Net is designed with an individualized, streamlined, low-complexity convolutional neural network, featuring dual CNN layers, interleaved with pooling layers, and a fully connected layer [16]. Dealing with real-world challenges of graphic or interactive data such as social networks, knowledge models, and molecular structures presents severe barriers to deep learning implementation due to their unique characteristics [17]. Altaee et al. [18] propose a multi-level feature fusion method combining remote sensing data with CNN-based image classification, improving classification accuracy and advancing the integration of diverse data sources in remote sensing applications. Mohamed et al. [19] proposes a deep learning framework that integrates CNNs with Attention mechanisms to enhance the early diagnosis of Alzheimer's disease, improving diagnostic accuracy through advanced analysis of medical imaging data.

Liao *et al.* [20] introduced a deep learning-based system for automated assessment of physical conditioning exercise habits. This framework included metrics to quantify movement performance, a scoring function to convert these metrics into numerical quality scores, and a deep neural network model that generated quality assessment through observational learning. Kanade *et al.* [21] proposed transformer-based architecture for movement quality assessment (MQA) was proposed in this area, with the addition of a focus to improve the accuracy and efficiency of movement quality scoring. Capecci *et al.* [22] developed a Hidden Semi Markov model (HSMM)-based method for evaluating and monitoring body movements during rehabilitative training programs, and combined rule-based and simulation-based methods for analysis a complete. Caru et al. [23] focused on the impact of rehabilitation programs on physiological and psychological outcomes, particularly in cardiac patients, and highlighted the continued popularity of exercise-based rehabilitation in the cardiovascular field the prevention of diseases. Mandolesi *et al.* [24] explored the biological and psychological benefits of physical exercise, examining brain plasticity and epigenetic mechanisms in both animals and humans, aiming to optimize the positive impacts of physical exercise while mitigating potential adverse effects such as exercise addiction.

Rahman *et al.* [25] introduced an automated method for identifying the types of therapeutic exercises performed by stroke patients during rehabilitation, and used the 3-layer CNN-LSTM model to provide accuracy and its effectiveness in detecting human activity increased. Schez-Sobrino *et al.* [26] developed the computer vision-based system was proposed for home use, enabling the automated assessment of physical activity using the data connections used in health care facilities, which included gaming techniques for patients. Iwao *et al.* [27] evaluated a long-term preventive care program combining exercise, oral health promotion, and nutritional counselling to improve the well-being of older adults in the community was evaluated. Alakkari et al. [28] explore advanced machine learning techniques for forecasting COVID-19 infection rates, utilizing encoder-decoder LSTM and attention LSTM algorithms. Their study highlights the effectiveness of these models in capturing complex temporal patterns in COVID-19 data, offering improved prediction accuracy. This work contributes to the broader literature on pandemic modeling, demonstrating how sophisticated deep learning approaches can enhance forecasting capabilities in public health scenarios. Myvizhi and Abdel-Monem [29] investigate the application of deep learning, specifically Long Short-Term Memory (LSTM) networks, for predicting wind turbine performance. Their research demonstrates how LSTM models can effectively handle the sequential data involved in wind energy forecasting, leading to predictions that are more accurate. This study adds to the growing body of literature on renewable energy, displaying the potential of deep learning techniques in enhancing the efficiency and reliability of wind power generation [30].

This research's principal contributions encompass the subsequent fundamental aspects:

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

- To select the optimal features, the new hybrid feature extraction model referred to as PutterFish Customized Serval Optimizer (PFCSO) is introduced.
- To introduce a new RehabNet++ architecture tailored for multimodal human movement assessment. The proposed RehabNet++ (proposed) includes the optimized MLP (proposed), Attention-based CNNs, STR-RESNET and Transfer learning, respectively.
- To enhance the detection accuracy, the hidden layers of MLP (proposed) are fine-tuned using the new hybrid optimization PutterFish Customized Serval Optimizer (PFCSO).
- To include Grad-CAM visualization to show graphical explanations for the model's conclusions.

The research follows a well-organized format: Section 1 introduces the research, Section 2 explores an overview of the literature, Section 3 provides specifics about the suggested methodology, Section 4 presents the findings and engages in discussions, and Section 5 summarizes the overall conclusions and results.

Based on the literature review, the problem statement emerges as follows: Despite significant advancements in utilizing deep learning and other computational techniques for assessing movement quality and rehabilitation effectiveness, current approaches still face challenges in achieving comprehensive, accurate, and accessible evaluations of physical rehabilitation exercises. Existing methods vary from deep learning-based frameworks and computer vision systems to rule-based models, yet gaps persist in integrating these diverse approaches into unified, efficient tools that can consistently monitor and optimize rehabilitation outcomes across different patient populations and clinical settings. This necessitates a focused effort towards developing robust, adaptable methodologies that can standardize assessment protocols, improve diagnostic accuracy, and enhance therapeutic interventions in physical medicine and rehabilitation.

3. Proposed Methodology for Physical Habitation via RehabNet++framework

3.1.1 RehabNet++framework

In this RehabNet++ framework shown in Figure 1, the journey begins with feature extraction, optimizing the input data's representation. Leveraging transfer learning augments this process. Next, the spotlight shifts to an attention-based Convolutional Neural Network that hones in on salient aspects of the feature set. The refined information then enters a STR-ResNet architecture for sequential pattern analysis. Following this, two dense layers further process the data, and the network converges through fully connected layers. Ultimately, a SoftMax layer provides predictive outcomes. With the addition of transfer learning and attention processes, this all-encompassing method produces extremely accurate predictions.

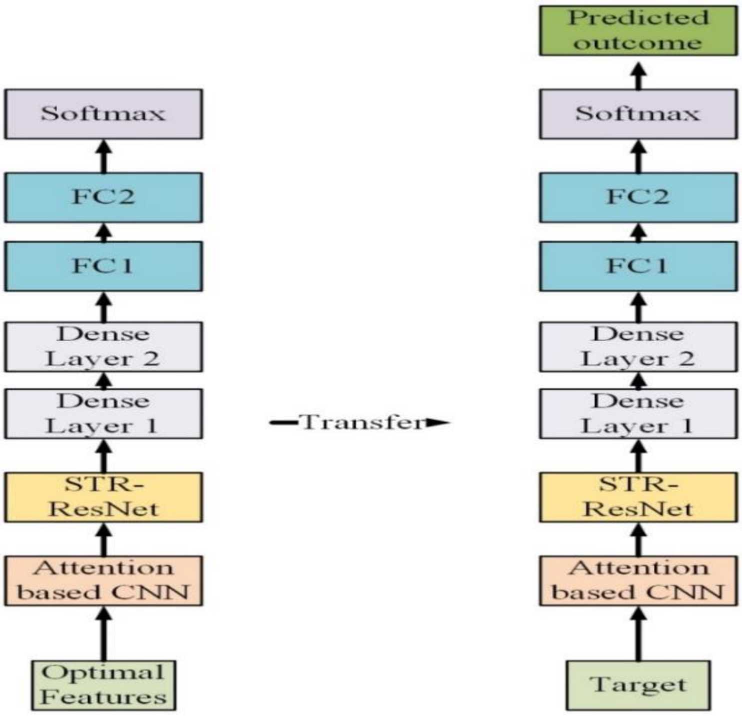


Figure 1. Flow diagram of Three Layer deep learning optimized MLP

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

Comprising thick layers that are fully linked, the Multi-Layer Perceptron, or MLP, is capable of reshaping input dimensions to fit any format that is required. There are connections between the neurons in this multi-layered neural network, which enables the outputs of certain neurons to function as inputs for other neurons. An MLP usually consists of an output layer with neurons matching to desired outputs, an input layer with one neuron per input, and an arbitrary number of hidden layers with different numbers of nodes.

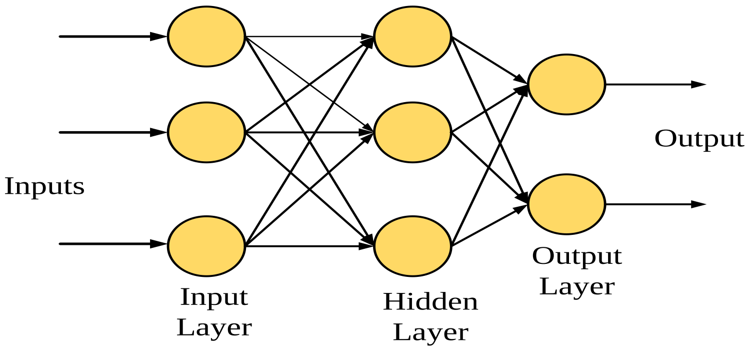


Figure 2. Optimized MLP

In the Figure 2. Three input nodes correspond to three inputs, a hidden layer with three nodes, and an output layer with two nodes for two outputs. Information flows from input nodes to hidden layer nodes and then to output nodes. Each node employs a sigmoid activation function, converting real values to values between 0 and 1 via the sigmoid formula. To improve detection accuracy, will fine-tune the hidden layers within the Multilayer Perceptron (MLP) using a novel hybrid optimization approach. This innovative method combines traditional training techniques with advanced optimization strategies to enhance the model's performance. By fine-tuning the hidden layers through this novel hybrid optimization model, aim to significantly boost the accuracy of the detection system.

3.1.1.1 Attention Based CNN

ABCNN as illustrated in Fig 3 uses an attention feature matrix denoted as a to influence its convolution process. The purpose of these attention features is to assign higher weights to those units in the left representation feature map S_0 that are relevant to units in the right representation feature map S_1 . S_{1-I} ($I \in \{0,1\}$) Here, Unit can indicate individual words or phrases at various network levels.

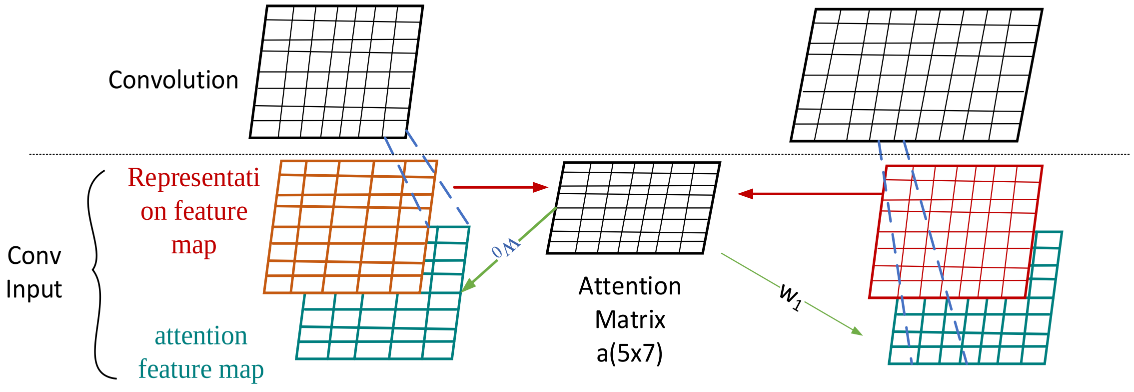


Figure 3. Block of ABCNN

This attention matrix A is generated by matching units in S_0 with units in S_1 . The values in a signify the attention distribution of each unit in S_0 concerning S_1 and vice versa. a Effectively serves as a novel feature map for both S_0 and S_1 . Each row or column corresponds to a feature vector of a unit in S_0 or S_1 . To harness this attention feature map optimally, it undergoes transformation into two blue matrices, aligning with the representation feature maps' format. Consequently, the convolution operation takes input from both the original representation feature maps (in red) and the transformed attention-based feature maps (in green). This concept allows the attention feature map to guide convolution in learning counterpart-biased sentence representations, thereby enhancing the model's capabilities.

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

Formally in Eq (1), for a sentence $I(I \in \{0,1\})$, the representation feature map $f_{I,R}$ is created in $r^{D\times S'}$. The attention matrix a is defined as follows in $r^{S\times S'}$.

$$a_{I,J} = match\_score(f_{0,R}[:,I], f_{1,R}[:,J]) \tag{1}$$

The match-score function can be defined using various methods, with a successful approach being $1/(1 + |X - Y|)$, where $|\ \cdot\ |$ represents the Euclidean distance are shown in Eq (2). Utilizing the attention matrix A, create the attention feature map $f_{I,A}$ for $S_I$as follows:

$$f_{0,A} = w_0 \cdot a^t, \quad f_{I,A} = w_1 \cdot a \tag{2}$$

The model's learnable parameters encompass weight matrices $w_0 \in r^{D\times S}$ and $w_1 \in r^{D\times S}$. During training, the representation feature map $f_{I,R}$ and the attention feature map $f_{I,A}$ are combined into an order-3 tensor. This tensor undergoes convolution, yielding an elevated-level representation feature map for each $S_I(I \in \{0,1\})$. In Figure 3, S_0 comprises 5 units, and S_1 has 7. The convolution's output, presented in the top layer with a filter width of $w = 3$,, results in a higher-level representation feature map with 7 columns for S_0 and 9 columns for S_1.

3.1.1.2 Spatiotemporal Convolution with Residual Connections (STR-ResNet)

The STR-ResNet architecture employs recurrent units to establish connections between multiple SRes-CNNs, enabling the embedding of temporal correlations. Unlike traditional approaches, STR-ResNet incorporates both low-resolution (LR) frames and the differences between adjacent LR frames as inputs. It reconstructs a high-resolution (HR) frame by merging the LR frame with the predicted spatial residue and considering the predicted temporal residues between adjacent frames. As depicted in Figure 4, STR-ResNet executes six specific operations to accomplish this fusion and prediction process. Forward Convolution: In each SResCNN component, convolution operations are performed for single-frame super-resolution.

- Convolution of Recurrent: To propagate information across recover lost details and adjacent frames, STR-ResNet conducts recurrent convolutions. These convolutions transfer features from the adjacent $(T - 1)th\ and\ (T + 1)th$ frames to the $T\ th$ frame.
- Context Convolution: Similar to recurrent convolutions, context convolutions transmit complementary information among frames. They propagate features from the adjacent $(T - 1)th\ and\ (T + 1)th\ frames\ to\ the\ T\ th$frame, enhancing contextual understanding.
- Temporal Residue Embedding: Within the 8th layer, predictions of temporal residues are made, governed by a loss function that guides them to regress towards the ground-truth temporal residues. Subsequently, these predicted residues are combined with the output feature maps from the 7th layer, resulting in the output feature maps of the 8th layer.
- Bypass of Feature: Incorporating features from the 1st and 4th layers, they merge with the output of the 3rd and 6th layers correspondingly, amplifying feature representation.
- LR Bypass: Directing LR frames to the output of the 8th layer produces the inferred HR details of frame T.
- Feed Forward: Propagating feature maps to the next unit via feed-forward procedures.

In the SRes-CNN architecture, the context convolutions and recurrent are selectively deployed, occurring solely within the $2nd, 3rd, 5th, and\ 6th\ layers$, Recurrent connections transmit outputs from these specific layers of the current frame (Tth) to their corresponding layers in the adjacent frames $(T-1)th\ and\ (T+1)th$. Conversely, context connections relay information from previous layers of the current frame (1st, 2nd, 4th, and 5th) to their corresponding subsequent layers in the adjacent frames $(T-1)th\ and\ (T+1)th$. After the convolutional operations, an element-wise summation amalgamates the outputs, resulting in a new feature map. These convolutions' outputs, along with the fusion output, significantly contribute to the model's representation learning process are given in Eq (3) - Eq (8).

$$c^{F}_{(T,I)} = \mathcal{W}^{F}_{I} * c^{A}_{(T,I-1)} + \mathrm{B}^{F}_{I} \tag{3}$$

$$c^{C,P}_{(T,I)} = \mathcal{W}^{C,P}_{I} * c^{A}_{(T-1,I-1)} + \mathrm{B}^{C,P}_{I} \tag{4}$$

$$c^{C,N}_{(T,I)} = \mathcal{W}^{C,N}_{I} * c^{A}_{(T+1,I-1)} + \mathrm{B}^{C,N}_{I} \tag{5}$$

$$c^{r,P}_{(T,I)} = \mathcal{W}^{r,P}_{I} * c^{A}_{(T-1,I)} + \mathrm{B}^{r,P}_{I} \tag{6}$$

$$c^{r,N}_{(T,I)} = \mathcal{W}^{r,N}_{I} * c^{A}_{(T+1,I)} + \mathrm{B}^{r,N}_{I} \tag{7}$$

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

$$c^{A}_{(T,I)} = \max(0, c^{F}_{(T,I)} + c^{C,P}_{(T,I)} + c^{C,N}_{(T,I)} + c^{r,P}_{(T,I)} + c^{r,N}_{(T,I)}) \tag{8}$$

In the context of the model architecture described, the operations are denoted by various superscripts and subscripts. The superscripts $F, C, r, and\ A$ denote different unit types: forward convolution, context convolution, recurrent convolution, and element-wise summation aggregation, respectively. Superscripts $P\ and\ N$ indicate the convolution direction, either from the previous or next frame. Subsequently, the subscript (T, I) signifies the operation on the $I - th$ layer of the $T - th$ frame. Thus, $c^{F}_{(T,I)}$, $c^{C,P}_{(T,I)}$, $c^{C,N}_{(T,I)}$, $c^{r,P}_{(T,I)}$, and $c^{r,N}_{(T,I)}$represent outputs of forward convolution, context convolution from the previous frame, context convolution from the next frame, recurrent convolution from the previous frame, and recurrent convolution from the next frame in the $I - th$ layer of the $T - th$ frame, respectively. $c^{A}_{(T,I)}$performs element-wise summation across all five outputs to integrate predictions from the current and adjacent frames. Following this aggregation, a Rectified Linear Unit activation function is applied in Eq (9).

$$c^{A}_{(T,I)} = c^{F}_{(T,I)}\ , for\ I = 1,4,7,9. \tag{9}$$

Transitioning to the 8th layer, emphasis is placed on forecasting temporal residues for HR frames. These predictions become integral elements of the features employed for spatial residue estimation. This methodology enriches the model's capability to grasp and depict temporal fluctuations within the data, consequently enhancing the precision of spatial residue estimation.

$$\delta^{X}_{T} = \mathcal{W}_{\delta} * c^{A}_{(T,7)} + \mathcal{B}_{\delta}, c^{A}_{(T,8)} = [c^{A}_{(T,7)}, \delta^{X}_{T}] \tag{10}$$

Through the integration of context and recurrent convolutions, coupled with constraints on temporal residues, STR-ResNet adeptly captures inter-frame motion context transmitted from neighboring frames for video super-resolution. This empowers the model to augment spatial resolution by harnessing temporal details, leading to enhanced video quality and sharper clarity.

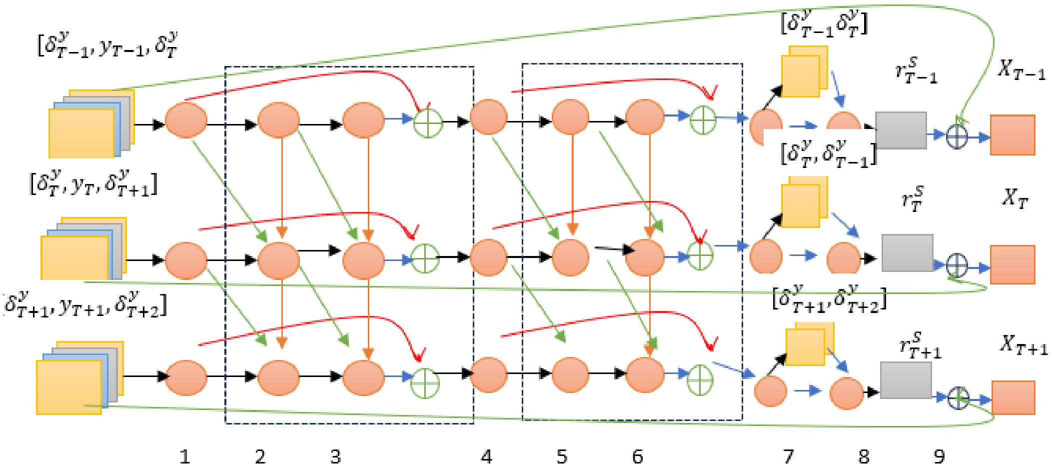


Figure 4. STR-ResNet architecture

3.1.1.3 Transfer Learning

A novel method that expands on the comprehension and analysis of human or animal motions is transfer learning for skeletal data. In this context, skeletons stand for the measured or inferred joint locations across time, which are frequently obtained via sensor or picture data. Transfer learning transfers the skills and characteristics that a pretrained model has acquired on a related task to a new but related challenge. Guaranteeing data consistency and quality through data gathering and preparation. Then, as a first step, a pre-trained model, usually a deep tissue is selected. This model is a valuable tool because it has already retrieved complex concepts and system characteristics from a large dataset during the initial training phase.

It changes the visualization to match the skeletal details. Final layers, originally developed for previously trained tasks, are added or added to other layers that can understand the details of the bones, for example, these layers can be placed in front of or around the joint need to be predicted if the goal is to approach the human level. Because

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

of these changes, the model can gradually become more adept at understanding and interpreting the skeletal details. Knowledge transfer forms the core of transfer learning. The weights from the pre-trained model are used to initialize the updated model, which is useful in obtaining relevant information. For more task-specific learning, you can fine-tune all layers or freeze the selection layer (feature extraction), depending on the amount of data available for the target task. Suitable loss functions should be defined and minimized by optimization methods for training the model. Continuous monitoring of the model's performance on the validation dataset allows adjustment of the hyperparameter for better results.

When dealing with raw data, transfer learning is crucial for skeletal data. It speeds up the learning process and produces models that are sensitive to a variety of skeletal motions in a unique and predictable way. This method is useful in many fields, including robotics, sports biology, and medical research, where a thorough grasp of skeletal motion is necessary for success.

- **Dense Layer**

The basis of neural networks is the complex component, sometimes called the fully integrated layer, that enables the understanding of data links and complex structures This layer enables complete exchange of information through subsequent layers of neural networks on the overlapping signals of all. Each connection in this category has a weight associated with it, and this weight is changed during training so that the layer can learn from the input. ReLU and sigmoid activation functions add nonlinearity, which improves the ability of the network to represent complex data combinations. Dense layers are important in applications such as image recognition and natural language processing to capture and transform input data into meaningful predictions or classifications

- **Fully connected Layer**

The fully connected layer is a crucial component of neural networks because it serves as a link between abstraction and data interpretation. If every neuron is in sync, with every other layer of neurons that come before and after it, this layer can communicate large amounts of data. During training, the correlation weights between neurons are changed to aid improve the network's understanding of intricate data structures. A fully connected layer uses activation functions to incorporate nonlinearity and enable the network to define complex relationships in data, such as ReLU or softmax. This layer is important for image recognition, natural language processing, processing while extracting the information necessary to enable accurate prediction and classification

- **SoftMax**

SoftMax is a crucial neural network component that is mostly utilized for multi-class classification problems. It translates raw output scores into class probabilities. One of its distinguishing characteristics is the normalization of scores, which ensures that the probability total up to one. Exponentiating the raw scores and dividing the result by the all exponentiated scores is how it does this. The result is a distribution where each class receives a probability score, conveying the probability of input belonging to a particular class. SoftMax plays a pivotal role in the decision-making process, determining the class with the highest probability. It finds extensive application in image classification, natural language processing, and various domains. The RehabNet++framework is trained with the selected optimal features among (Joint Level Features, Limb-Level Features, Temporal Features and Statistical Features) acquired from PutterFish Customized Serval Optimizer (PFCSO) model.

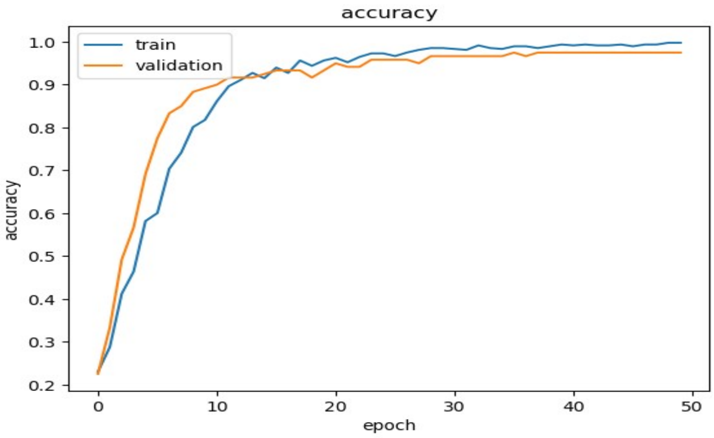


Figure 5. Graphical representation of the accuracy with epoch of the train and validation

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

Figure 5 illustrates the graphical representation of the accuracy with epoch of the train and validation of the proposed model. It is clear that the training and validation of the accuracy with epoch is gradually increases due to the hybrid optimization algorithm.

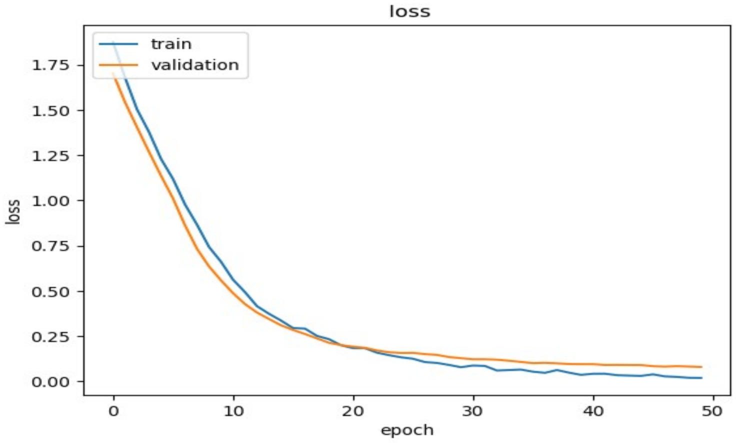
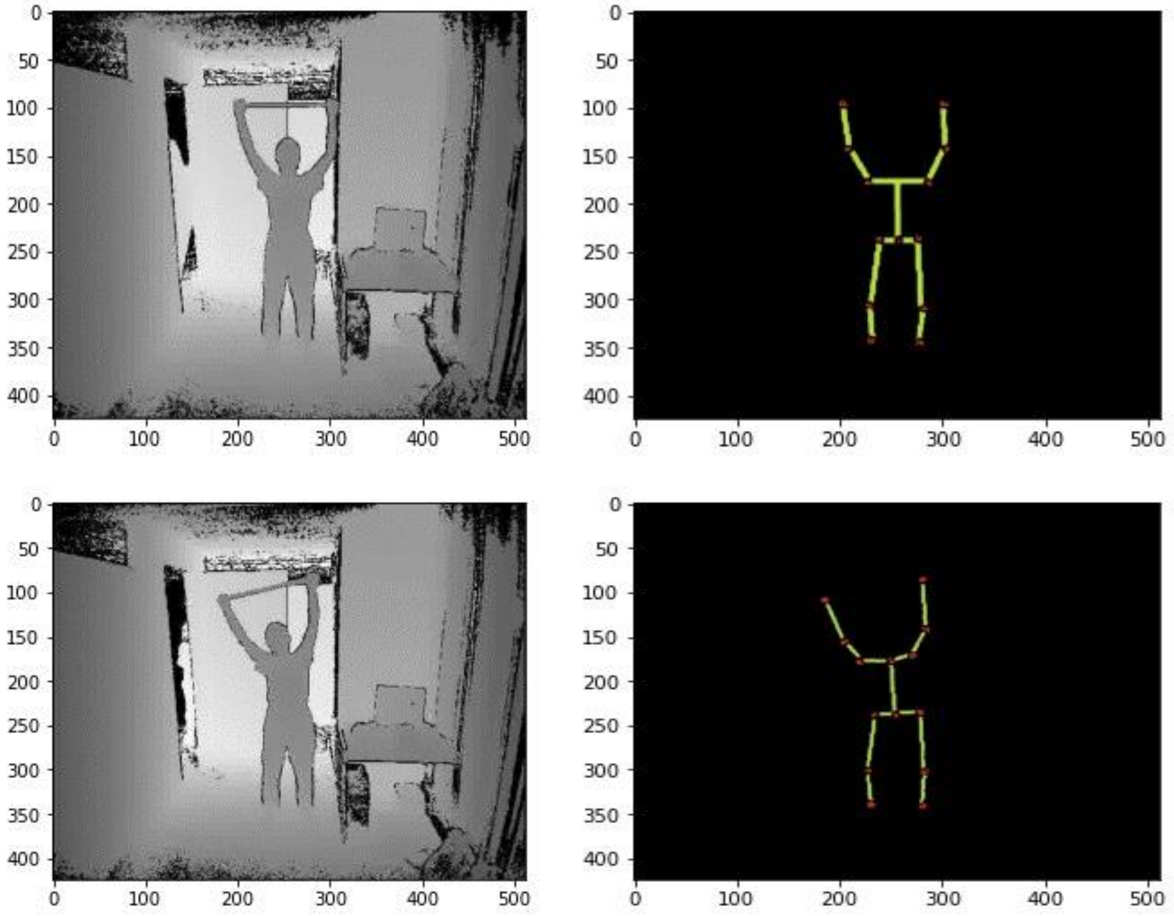


Figure 6. Graphical representation of the loss with epoch of the train and validation of the proposed model

Figure 6 illustrates the graphical representation of the loss with epoch of the train and validation of the proposed model. From the above figure, it is clear that the loss value due to train and validation is gradually decreases to 0.00, because of the usage of the hybrid optimization algorithm.



DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

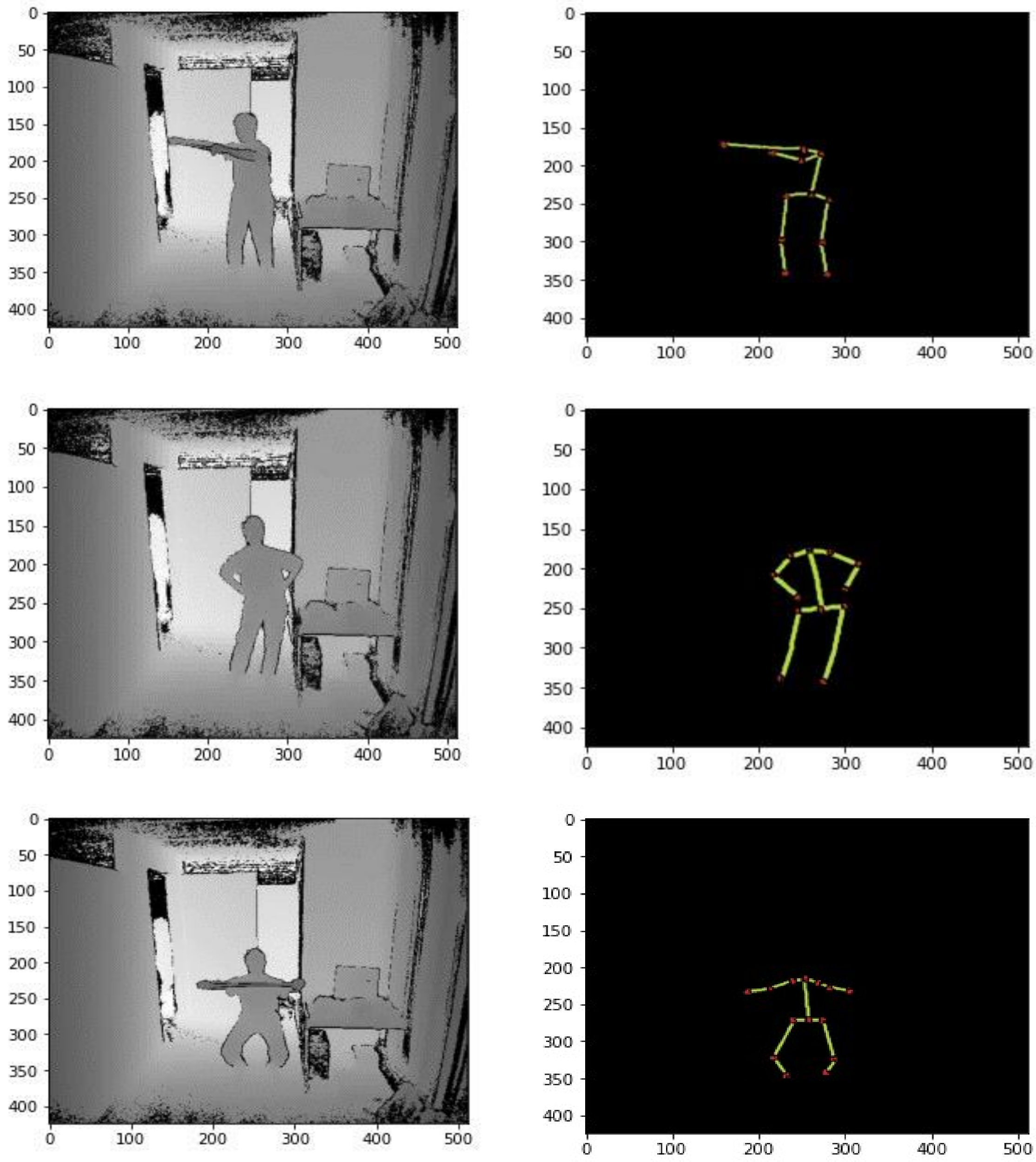


Figure 7. Illustrates the Input images of EX1, EX2, EX3, EX4, EX5 and its corresponding feature extraction images.

3.2.1 Joint Level Features

In image processing, joint-level features pertain to attributes or characteristics extracted from pairs or groups of pixels or image regions, rather than individual pixels. These features capture relationships, patterns, and interactions between different parts of an image. For example, co-occurrence matrices are used to compute texture features by analyzing pixel pairs' spatial relationships. Joint histograms illustrate the links between the color channels in color images. These joint-level characteristics are essential because they capture more accurate information about the content and structure of images, which is useful for tasks like object recognition, texture analysis, and picture segmentation.

3.2.2 Limb-Level Features

Legs or other body components of a human or animal subject are examples of features that may be taken out of particular regions or structures in a picture and applied to image processing. Because these characteristics concentrate on fine-grained details inside the image, they are utilized for tasks like object detection and posture prediction.

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

3.2.3 Temporal Features

Temporal features are information or patterns that change over time in the context of deep learning. They are essential for tasks like time-series forecasting, voice recognition, and video analysis. Models that reflect these temporal correlations, such as transformers and the STR-ResNet architecture, can aid understanding and predicting sequential data.

3.2.4 Statistical Features

- **Mean and variance**

Divide the total number of pixels by the values to get the pixel sum. The brightness or hue of every pixel in a picture recorded on a computer is information. The mean value represents how each pixel contributes to the overall intensity of the image, while variance is typically used to determine how each pixel differs from the center pixel or nearby pixels and is used to categorize the image into distinct regions.

$$\mu = \frac{1}{N}\sum_{1}^{N} X_I \tag{11}$$

$$Variance = \frac{(\sum(X-\mu)^2)}{n} \tag{12}$$

Where the number of terms in distribution is represented as n; μ is the mean shown in Eq (11).

- **Range**

The range of your data in statistics is the range from the distribution's lowest value to its highest value. It serves as a typical indicator of unpredictability. Measurements of variability provide descriptive statistics for your data set's summary, together with measurements of central tendency. The formula are shown in Eq (13).

$$R = \max(x_1, x_2, \dots . x_N) - \min((x_1, x_2, \dots . x_N) \tag{13}$$

R refers the range;

- **Standard Deviation of Joint angles**

The Standard Deviation of joint angles in image processing is a statistical measure that quantifies the variability or dispersion of joint angles within a set of data points, often used in tasks like pose estimation. The formula for calculating the standard deviation (σ) of a set of joint and limb angles (θ) are shown in Eq (14):

$$\sigma = \sqrt{\frac{1}{n}\sum_{I=1}^{n}(\theta_I - \mu)^2} \tag{14}$$

Where:

n is the number of joint and limb angle measurements.

θ_I represents individual joint and limb angle values.

μ is the mean (average) of the joint and limb angle values.

This formula calculates the spread or deviation of joint and limb angles from their mean, providing insight into the overall variability of the pose.

3.3.1 PutterFish Customized Serval Optimizer (PFCSO)

The PutterFish Customized Serval Optimizer (PFCSO) model is the combination of Pufferfish and Serval algorithms. With PFCSO, the optimal features among Joint Level Features, Limb-Level Features, Temporal Features and Statistical Features have been extracted.

Step 1: Initialization

The POA method employs a population-based strategy to solve optimization challenges iteratively. Each member represents a potential solution, forming a matrix of vectors in the problem space. Equation (16) initializes their positions, crucial for subsequent exploration. This approach harnesses collective search power for effective optimization.

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_I \\ \vdots \\ x_n \end{bmatrix}_{n\times M} = \begin{bmatrix} X_{1,1} & \cdots & X_{1,D} & \cdots & X_{1,M} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ X_{I,1} & \cdots & X_{I,D} & \cdots & X_{I,M} \\ \vdots & \iddots & \vdots & \ddots & \vdots \\ X_{n,1} & \cdots & X_{n,D} & \cdots & X_{n,M} \end{bmatrix}_{n\times M} \quad (15)$$

$$X_{I,D} = LB_D + R \cdot (UB_D - LB_D) \quad (16)$$

In this context, X represents the matrix encompassing the POA population, where each Xi signifies the $I - th$ member, and $X_{I,D}$denotes its d-th dimension within the search space. With n as the population size and m as the decision variables count, random number $r \in [0, 1]$ influences initialization. Lower & upper bounds LB_D and UB_D constrain variable ranges.

SOA introduces a population-based optimization method, drawing inspiration from servals' predatory behaviour. Servals represent the population of search agents, each embodying a candidate solution as a vector in the search space. Equation (18) initiates their positions, crucial for subsequent exploration within the optimization process.

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_I \\ \vdots \\ x_n \end{bmatrix}_{n\times D} = \begin{bmatrix} X_{1,1} & \cdots & X_{1,J} & \cdots & X_{1,D} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ X_{I,1} & \cdots & X_{I,J} & \cdots & X_{I,D} \\ \vdots & \iddots & \vdots & \ddots & \vdots \\ X_{n,1} & \cdots & X_{n,J} & \cdots & X_{n,D} \end{bmatrix}_{n\times D} \quad (17)$$

$$X_{I,J} = LB_J + R_{I,J} \cdot \left(UB_J - LB_J\right), I = 1,2, \dots, n \text{ } and \text{ } J = 1,2, \dots, D \quad (18)$$

In this context, x represents the matrix containing the locations of servals, where x_I denotes the $I - th$ serval, and $X_{I,J}$ signifies its $J - th$ the search space dimension, corresponding to a variable. n represents the total count of servals, while D indicates the count of decision variables. Random numbers $R_{I,J}$ from the interval [0,1] influence the algorithm's randomness, while LB_J and UB_Jdenote the lower and upper bounds, respectively, for the $J - th$ decision variable.

- Evaluate the objective function values for each candidate solution

$$f = \begin{bmatrix} f_1 \\ \vdots \\ f_I \\ \vdots \\ f_n \end{bmatrix}_{n\times 1} = \begin{bmatrix} f(x_1) \\ \vdots \\ f(x_I) \\ \vdots \\ f(x_n) \end{bmatrix}_{n\times 1} \quad (19)$$

In this context, f signifies the vector containing the assessed values of the objective function. where f_I $r$epresents the objective function evaluated based on the I-th POA member. This vector encapsulates the performance metrics of each candidate solution in the optimization process.

As each serval represents a candidate solution, the objective function assesses his or her proposed values for decision variables. Equation (19) then condenses these evaluations into a vector, portraying the objective function's values across the serval population. This vector provides insight into the performance of each candidate solution within the optimization framework.

Step 2: Exploration phase

During the initial POA phase, population members simulate predator attacks on pufferfish, leveraging the latter's slow speed as easy prey. Position updates mimic predator movements, fostering exploration for global search. Each member, acting as a predator, targets pufferfish with superior objective function values, identified through Equation (20).

$$cp_I = \{x_K : f_K < f_I \text{ } and \text{ } K \neq I\}, where \text{ } I = 1,2 \dots n \text{ } and \text{ } K \in \{1,2, \dots, n\} \quad (20)$$

In the POA design, cp_I represents the potential locations of pufferfish for the $I - th$ predator, where x_K signifies a population member with a higher objective function value than the $I - th$ predator, and $f_K$denotes its corresponding objective function value.

$$X_{I,J}^{sp1} = X_{I,J} + R_{I,J} \cdot (sp_{I,J} - I_{I,J} \cdot X_{I,J}) \quad (21)$$

$$x_I = X_{I,J}^{p1} \begin{cases} x_I^{p1}, f_I^{p1} \le f_I \\ x_I, else, \end{cases} \quad (22)$$

The predator randomly selects a pufferfish from cp_I as $sp_{I,J}$. Utilizing Equation (21), each POA member computes a new position based on predator-pufferfish dynamics. If the objective function improves at this new position, Equation (22) adjusts the member's coordinates. Here, $sp_{I,J}$represents the randomly selected pufferfish for the $I-th$ predator from the set cp_I, with $sp_{I,J}$ indicating its J-th dimension. X_I^{p1} signifies the newly computed position for the I-th predator in the first POA phase, where X_J^{p1}denotes its $J-th$ dimension. f_I^{p1}indicates its objective function value. $R_{I,J}$ denotes random numbers within the range [0,1], and $I_{I,J}$ are randomly chosen as either 1 or 2.

Within the SOA framework, the population's top performer is designated as the prey, guiding the serval's pursuit. Equation (21) computes the serval's new position mimicking the attack, aiming to enhance the objective function value. Equation (22) updates the serval's position if improvement occurs, optimizing its hunting strategy.

$$X_{I,J}^{p1} = X_{I,J} + R_{I,J} \cdot \left(p_J - I_{I,J} \cdot X_{I,J}\right), I = 1,2 \ldots, n \; and \; J = 1,2, \ldots, D \qquad (23)$$

Step 3: Exploitation phase

- Update the position of each Serval based on the chase process using Equation (24).

$$X_{I,J}^{sp2} = X_{I,J} + \frac{R_{I,J} \cdot (UB_J - LB_J)}{T}, I = 1,2 \ldots n, J = 1,2 \ldots D \; and \; T = 1,2 \ldots, t \qquad (24)$$

X_I^{sp2} represents the updated position calculated for the $I-th$ predator, with $X_{I,J}^{sp2}$ indicating its J −th dimension. Random numbers between 0 and 1 are denoted as $R_{I,J}$, while T represents the iteration count.

- If the value of objective function improves, serval position is updated by using the Equation (25).

$$x_I = X_{I,J}^{sp2} \begin{cases} x_I^{sp2}, f_I^{sp2} \le f_I \\ x_I, else, \end{cases} \qquad (25)$$

- In a similar vein, use Equation (26), to update each member's location for the Pufferfish method.

$$X_{I,J}^{p2} = X_{I,J} + (1 - 2R_{I,J}) \cdot \frac{UB_J - LB_J}{T} \; (26)$$

- Equation (27), if the goal function's value increases, is used to update the member position.

$$x_I = X_{I,J}^{p2} \begin{cases} x_I^{p2}, f_I^{p2} \le f_I \\ x_I, else, \end{cases} \qquad (27)$$

Step 4: Termination

- Repeat the iteration process until a termination criterion is met.

Through the integration of the exploration and exploitation approaches of the Pufferfish and Serval algorithms, we can effectively capitalize on their individual advantages to thoroughly comb through the search space and identify optimum solutions by taking advantage of attractive locations. Because of the decentralized structure of the blockchain, which offers transparency and security throughout the optimization process, running these algorithms on the blockchain also guarantees data privacy and integrity. They may create the hybrid part of the algorithm by combining their search tactics to take use of their advantages in both the exploitation and exploration phases:

$$x_{hybrid} = \alpha \cdot x_{poA} + (1 - \alpha) \cdot x_{soA} \qquad (28)$$

x_{hybrid} Represents the hybridized positions of the solutions in Eq (28).

- x_{poA} Represents the positions of solutions from the POA algorithm.
- x_{soA} Represents the positions of solutions from the SOA algorithm.
- α Represents the weighting factor, determining the contribution of each algorithm. Typically, α is a value between 0 and 1.

This hybridization technique allows for a more robust optimization process by combining the exploration capabilities of one algorithm with the exploitation capabilities of the other, potentially leading to improved convergence and solution quality. Adjusting the weighting factor α enables fine-tuning of the balance between exploration and exploitation according to the specific characteristics of the optimization problem.

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

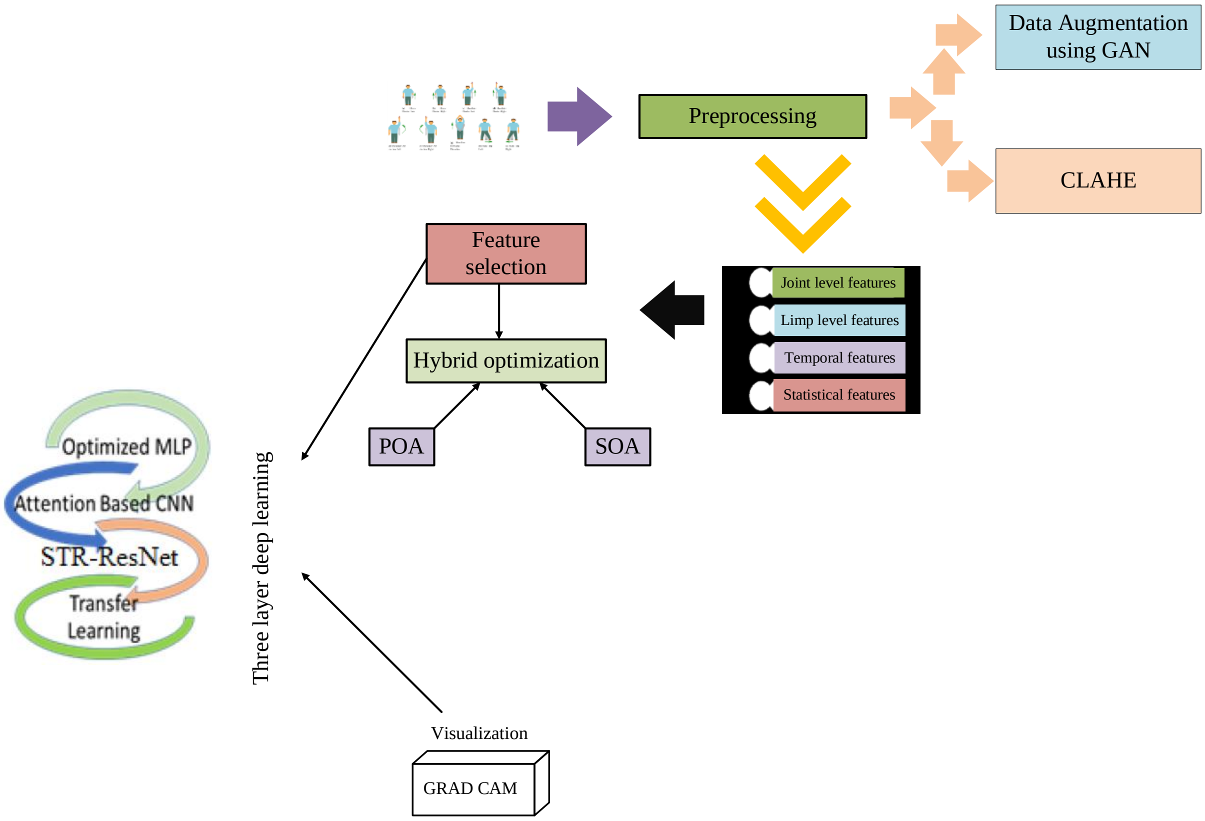


Figure 8. Architecture of proposed physical habitation model

3.4 Image Preprocessing

The first phase of this strategy involves guaranteeing the uniformity of skeleton data joint position images, then diving into the utilization of cutting-edge GAN-driven skeleton augmentation methods to expand the dataset. Furthermore, use the CLAHE technique to elevate image quality, with the ultimate goal of enhancing model generalization and overall performance.

3.4.1 Data augmentation

Data augmentation is a method that involves expanding the training dataset artificially by generating altered versions of existing data. This can involve introducing subtle variations to the original dataset or employing deep learning techniques to produce entirely new data points, ultimately enriching the training set for improved model performance.

- **Skeleton Augmentation using GAN**

Only a limited amount of alternative data is generated by conventional augmentation methods. Additionally, Generative Adversarial Networks (GANs) are used for data augmentation in which they produce copies of the original image and which can be used to create a big data set. Without explicated training data, GANs can learn deep representations. GANs are made up of two neural networks, the discriminator and the generator, as depicted in Figure (2). The Generator (G) will try to generate the replica while learning from the data distributions provided. The Discriminator (D) will determine whether the generated copy of data is real or fake, that is, whether the input data comes from the genuine training dataset or from the generator. When the Discriminator finds the fake data. It will be returned to the generator. The Generator will attempt to provide data that are more realistic after learning from the loss function. The process will repeat until the Discriminator is unable to tell the difference between actual and fake data.

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

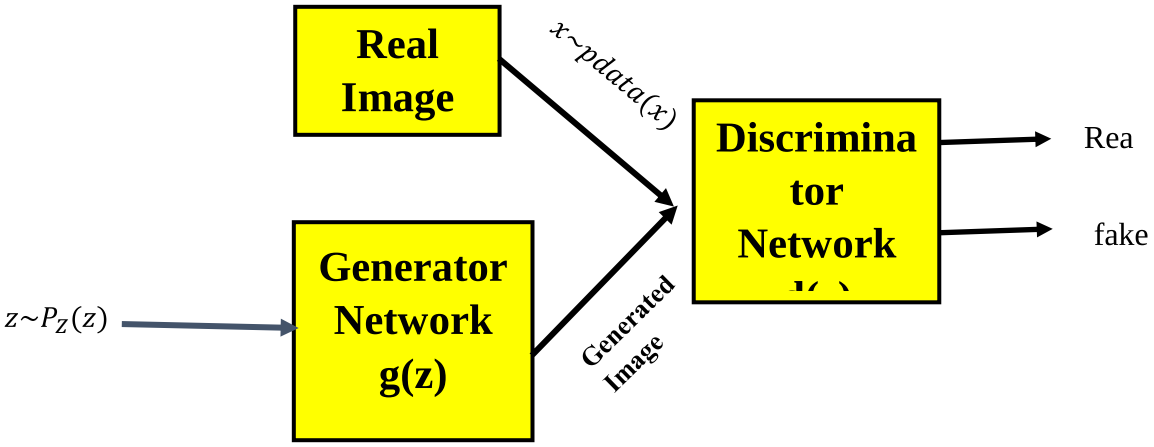


Figure 9. Architecture of GAN

3.4.2 CLAHE

The Contrast Limited Adaptive Histogram Equalization method processes localized image sections, referred to as tiles, rather than the entire image. It alleviates artificial seams between adjacent tiles by employing bilinear interpolation. This technique is particularly effective in enhancing image contrast. For color images, it's typical to apply this method to the luminance channel in the HSV color space. Remarkably, equalizing only the luminance channel within an HSV image often outperforms equalizing all channels in a standard BGR image.

4. Result and Discussion

4.1 Experimental Setup

In this research, to get the best accuracy the proposed model is implemented in PYTHON with KIMORE dataset. Ex1, Ex2, Ex3, Ex4, Ex5 are the five types of Physical rehabilitation exercises in the KIMORE dataset. In Ex1, the individual grasps a bar, extending their arms alongside the body. In the first exercise, participants gradually raise their arms above their heads, keeping their elbows extended to stretch the trunk muscles. Their feet remain grounded, knees slightly bent, and they maintain a neutral pelvis position for form and effectiveness. Exercise 2 involves lifting the arms above the head with fully extended elbows while holding a bar with both hands (initial position). The participants gently tilt their trunk to the left and then to the right, ensuring it stays aligned with the frontal plane. They return to the initial position, focusing on avoiding backward or forward trunk bending. Feet are planted on the ground, slightly apart, with slightly flexed knees for stability. Exercise 3 focuses on trunk rotation. Beginning with arms extended at a 90-degree angle from the torso and fully extended elbows, participants gently rotate their torso first to the left and then to the right. After the rightward rotation, they return to the starting position while maintaining proper alignment without leaning forward or backward. Feet are firmly grounded, slightly apart, with knees slightly flexed for stability. In Exercise 4, individuals stand with their feet slightly apart and remain stationary. Without shifting their feet, they perform circular rotations of the pelvis, starting with clockwise rotations and then switching to counterclockwise rotations. This exercise enhances pelvis mobility within the transverse plane while maintaining a stable stance. Exercise 5 involves squatting. Beginning with arms aligned at a 90-degree angle to the trunk and fully extended elbows (initial position), participants bend their knees to approximately 60 to 70 degrees and then return to the starting position. Throughout, they must maintain proper alignment in the sagittal plane, preventing any forward or backward trunk bending.

The computational time (ms) for the proposed model training set is 534 and testing set is 543.

4.2 Overall performance analysis of the suggested model and other comparable models

Table 1 provides a comprehensive overview of the performance metrics for the Hidden Semi-Markov Model (HSMM [18]). The model highlighted strong accuracy at 0.929, highlighting its capacity to make correct classifications. Precision values consistently exceeded 0.88, with the highest precision at 0.978, indicating the model's ability to minimize false positives. The recall and sensitivity metrics were solid, ranging from 0.853 to 0.969, emphasizing the model's effectiveness in capturing true positives. The F1-scores revealed a balanced trade-off between recall and accuracy, remaining high and averaging at about 0.955. HSMM [18] demonstrated

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

outstanding specificity, with the smallest value being 0.88, showing its capability of correctly identifying negative cases. With an FPR ranging from 0.057 to 0.089 and a relatively low FNR of 0.06 to 0.089, the model manifested itself by reducing errors in classification. The Matthews Correlation Coefficient (MCC) of 0.829–0.96 values supported the non-elite credibility of the model in numerous classes of categorization scenarios. A range of categorization features perform well for the HSMM [18] model, such as the combined metrics shown.

Table 1: Overall performance of the metrics of the model HSMM [18]

	HSMM [18] (Hidden Semi-Markov Model)				
Accuracy	0.929	0.96	0.955	0.87	0.923
Precision	0.959	0.968	0.959	0.88	0.953
Recall	0.929	0.969	0.93	0.87	0.853
sensitivity	0.929	0.969	0.93	0.87	0.853
F1-score	0.959	0.96	0.955	0.87	0.923
Specificity	0.959	0.978	0.959	0.88	0.963
NPV	0.929	0.96	0.94	0.84	0.933
MCC	0.925	0.96	0.92	0.829	0.903
FPR	0.057	0.066	0.081	0.089	0.065
FNR	0.067	0.06	0.081	0.089	0.065

Table 2 displays the overview of all the detail performance indicators for the five trials (Ex1 through Ex5) of the LSTM model [21]. In all these studies, there is a very high performance of the model consistently. The findings shown by LSTM [21] demonstrate its capability for accurate data classification as evident in high accuracy rates that vary between 87% and 97%. While sharing this same level of precision with other works at [92%-97%], implying that it is good at detecting instances of positive cases; moreover, the recurrent model has good recall and sensitivity values of above 87% in most cases, which means it can identify true positive instances.

In most analyses, the F1-score (which balances recall and precision) remains consistently high at 90%. The best-performing model development would yield from 86% to 98% specificity—the ability to find out true negatives. Negative Predictive Value (NPV) is another way of showing how good the model is at predicting the non-occurrence of negative events, with values always over 84% as an indication that it would correctly predict when things are likely to go wrong. The Matthews correlation coefficient (MCC) values indicate that the model is performing well on average, with values ranging between 83% and 97%; the dependability of the model is backed up by low false positive or negative rates, whereas a good proportion of different studies have often registered improved accuracy, precision, recall, specificity, and general effectiveness during classification tasks through the application of the LSTM [21] model.

Table 2: Overall performance of the metrics of the model LSTM [21]

	LSTM [21]				
	Ex1	**Ex2**	**Ex3**	**Ex4**	**Ex5**
Accuracy	0.97	0.96	0.955	0.87	0.923
Precision	0.967	0.97	0.955	0.92	0.97
Recall	0.937	0.97	0.935	0.89	0.87
sensitivity	0.937	0.97	0.935	0.89	0.87
F1-score	00.957	00.97	00.945	00.9	00.91
Specificity	00.977	00.97	00.979	00.86	00.98

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

NPV	00.93	00.97	00.948	00.849	00.94
MCC	00.93	00.965	00.93	00.83	00.9
FPR	0.05	0.06	0.047	0.079	0.07
FNR	0.057	0.05	0.077	0.085	0.05

Table 3 in each of the five unique trials (Ex1 through Ex5), Table 3 provides a comprehensive overview of the performance indicators of the CNN [14] model. According to the findings from these experiments, the CNN [14] model consistently offers favorable performances; hence, it is useful in various classification tasks. The CNN [14] model is well known for delivering accurate classifications. The accuracy rates are always on point since most fall within 90% to 98%, referring to how CNN [14] is able to pinpoint data. The remarkably high precision values exhibited in a number of studies demonstrate that its accuracy in accurately spotting positive cases is 93–99%. With good recall and sensitivity values of mostly 88% and above, it can consistently detect real positive instances.

Most studies have reported that the F1-score had a significant value of 91% in terms of recall and precision for most runs. The trade-off between specificity, which assesses how good the model is at identifying true negatives, ranged from good to better, with lower values being 87% and higher ones amounting up to 99%, whereas the negative predictive value (NPV), an indicator reflecting the model's ability to predict bad outcomes, was consistently above 85%, as shown by numerous trials. The incredible values of the Matthews Correlation Coefficient (MCC) between 83% and 97% illustrate the general terrific performance of the model. Besides proving that this model is dependable and has fewer false positive and negative rates, its capabilities in classification have been proven through various trials, as it is always accurate, precise, selective, and effective, hence making it also remain stable across different datasets [14].

Table 3: Overall performance of the metrics of model CNN [14]

	CNN [14]				
Metrics	**Ex1**	**Ex2**	**Ex3**	**Ex4**	**Ex5**
Accuracy	0.969	0.98	0.965	0.9	0.95
Precision	0.979	0.99	0.975	0.93	0.98
Recall	0.949	0.99	0.945	0.9	0.88
sensitivity	0.949	0.99	0.945	0.9	0.88
F1-score	00.969	00.99	00.965	00.91	00.92
Specificity	00.989	00.99	00.975	00.87	00.99
NPV	00.933	00.996	00.95	00.85	00.95
MCC	00.939	00.97	00.941	00.832	00.913
FPR	0.037	0.03	0.02	0.063	0.06
FNR	0.049	0.03	0.06	0.083	0.03

An evaluation of five different tests is shown in Table 4 by the use of performance indicators for the model in question. The model is effective when applied to different classification tasks and repeatedly shows very good results; this is evident from the performance of the PutterFish Customized Serval Optimizer (PFCSO). There is clear evidence that the model has excellent performance in terms of classifying data, judging from its consistent high accuracy scores recorded at 90.4% all through 100%. This shows that it is very good at determining which ones belong to positive examples. The model maintains high recall and sensitivity values that are also usually above 89%, indicating that it has a good capacity for catching real positive cases. Most studies have a high "F1-score," which is about 93%, showing the trade-off that exists between recall and precision. In some cases, this model's specificity in detecting true negatives can hardly be matched. A consistent demonstration of how good the model can be at forecasting negative happenings is found in the negative predictive value (NPV), which always

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

maintains above 85%. Generally, it is ranked as a very good model because its MCC values are often greater than 84%. You also observe that the consistent low rates of FP and FN further provide evidence of how reliable or resilient such models can be. Therefore, due to its amazing accuracy, precision, recall, specificity, and overall effectiveness across various trials consistently, this could still be regarded as the best model to use for any classification purposes.

Table 4: Overall performance of the metrics of the proposed model (CPOA)

	Proposed (CPOA)				
	Ex1	**Ex2**	**Ex3**	**Ex4**	**Ex5**
Accuracy	0.972	1	0.975	0.904	0.966
Precision	1	1	1	0.952	1
Recall	0.952	1	0.95	0.909	0.894
sensitivity	0.952	1	0.95	0.909	0.894
F1-score	0.975	1	0.974	0.9302	0.944
Specificity	1	1	1	0.894	1
NPV	0.939	1	0.952	0.859	0.952
MCC	0.945	1	0.951	0.842	0.923
FPR	0.01	0.01	0.01	0.05	0.05
FNR	0.047	0.01	0.05	0.08	0.02

Table 5 shows an all-round assessment of the GCN (Graph Convolutional Networks) model [2] performance. The ability of this model is measured using several performance measures on five different scenarios labelled as Ex1, Ex2, Ex3, Ex4, and Ex5. Initially, GCN's (Graph Convolutional Networks) high accuracy is evident from its first very high score; 0.955 for Ex1, 0.953 for Ex2, 0.931 for Ex3, 0.852 for Ex4, and 0.913 for Ex5. The model can categorize cases and it is able to do that quite well. Notably, Ex1 and Ex2 depict better results among others. The model was very accurate when it comes to making positive predictions in all five cases, having its precision scores range from 0.953-0.961, 0.932-0.95 and 0.943 respectively. This especially applies to Ex2, implying that it has the best accuracy. The machine showed good memory and sensitivity capacities by getting 0.955, 0.951, 0.933, 0.852 and 0.863 values consistently. In each of these measurements, Ex3 has the highest recall and sensitivity revealing a recognition capability for true positive cases of GCN [2]. Combining accuracy with recall yielded F1-score values 0.955, 0.961, 0.931, 0.852 and 0.913 in all situations thus showing that it is consistently robust.

Presumably, we may argue that Ex2 presented a balanced model performance with a high F1-score. Its scores expressed high specificity, which assesses how well a model, can detect negative examples: 0.963, 0.964, 0.935, 0.85, and 0.973. The model was particularly good at distinguishing bad cases. Negative predictive value (NPV) values of 0.952, 0.931, 0.832, and 0.913 invariably showed the model's accuracy in recognizing real negative cases in all situations. Ex2 had the greatest NPV. With MCC values of 0.92, 0.953, 0.911, 0.825, and 0.903.

The model performed considerably well. It is obvious that Ex2 has the highest MCC score. Besides, specifically, in the case with different FPR and FNR values of 0.077, 0.073, 0.087, 0.097, and 0.071 and 0.0777, 0.072, 0.087, 0.095, and 0.075 sequentially the model depicted consistently lower values of false positive rate (FPR) together with false negative rate (FNR). These low rates show that the GCN [2] model reduces errors for both positive and negative outcomes, with Ex5 having the lowest FPR and FNR. In conclusion, GCN [2] model achieves good results on many evaluation criteria which points out its use in multiple situations. Ex2 and Ex5 had very good results.

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

Table 5: Overall performance of the GCN [2] model

	GCN [2] (Graph convolutional networks)				
	Ex1	**Ex2**	**Ex3**	**Ex4**	**Ex5**
Accuracy	0.955	0.953	0.931	0.852	0.913
Precision	0.953	0.961	0.932	0.85	0.943
Recall	0.955	0.951	0.933	0.852	0.863
sensitivity	0.955	0.952	0.933	0.852	0.863
F1-score	0.955	0.961	0.931	0.852	0.913
Specificity	0.963	0.964	0.935	0.85	0.973
NPV	0.925	0.952	0.931	0.832	0.913
MCC	0.92	0.953	0.911	0.825	0.903
FPR	0.0774	0.073	0.087	0.097	0.071
FNR	0.0777	0.072	0.087	0.095	0.075

Consider five distinct possibilities: Ex1, Ex2, Ex3, Ex4, and Ex5. Table 6 contains a detailed analysis of the overall performance of the DTW model [3] using different performance indicators. For one thing, in terms of accuracy, the DTW model [3] performed well on a consistent basis. In Ex1 it had an accuracy of 0.901, then Ex2 0.93, Ex3 0.901, Ex4 0.859, and Ex5 0.901. This shows that the model was able to correctly classify events within those contexts (Ex2 was exceptionally correct). The model's performance was also fierce, as it recorded 0.939, 0.95, 0.927, 0.859, and 0.9 for every instance with respective relevance scores. Ex2, with high precision scores, recorded the highest precision, suggesting that the model made dependable positive forecasts. Sensitivity, also known as recall values of interest, was consistently maintained, implying the model is good at identifying real positive instances. The DTW [3] attained recall figures 0.901, 0.9301, 0.902, 0.859, and 0.83 in different circumstances recorded; the highest recall was observed in example two. The F1-score—, which is a balance between precision and recall—remained relatively constant at 0.93, 0.9301, 0.92, 0.859, and 0.9, supporting that the model can maintain consistent ratios of true positive estimates to positive cases generally. However, the model was excellent at recognizing negative cases with specificity values such as 0.939, 0.9502, 0.927, 0.859, and 0.95. The negative predictive value showed that our model is able to correctly distinguish incorrect states, with Ex5 having the highest specificity. The negative predictive value (NPV) was substantially high, which means that the model could pinpoint genuine cases of negativity with accuracy rates as follows: 0.903, 0.9301, 0.901, 0.819, and 0.901. Ex2 had the highest net present value (NPV) from the sample. The Matthews Correlation Coefficient (MCC) at 0.905, 0.93, 0.902, 0.82, and 0.89, taking into consideration real and false positives and negatives, further demonstrated the model's overall efficacy. The false positive rate (FPR) and false negative rate (FNR) both remained consistently low, with FPR values of 0.0794, 0.089, 0.099, and 0.09 and FNR values of 0.079, 0.0793, 0.089, 0.099, and 0.09. These low rates serve as the basis for the model being reliable enough, as they demonstrate that it can lower errors either way. In conclusion, the DTW [3] model presented consistent outcomes under variable evaluation parameters, meaning it could work in many different environments. Remarkably, Ex2 had remarkably good results.

Table 6: Overall performance of the DTW [3] model

	DTW [3] (Dynamic Time Wrapping)				
	Ex1	**Ex2**	**Ex3**	**Ex4**	**Ex5**
Accuracy	0.901	0.93	0.901	0.859	0.9
Precision	0.939	0.95	0.927	0.859	0.9
Recall	0.901	0.9301	0.902	0.859	0.83
sensitivity	0.901	0.9301	0.902	0.859	0.84

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

F1-score	0.93	0.9301	0.92	0.859	0.9
Specificity	0.939	0.9502	0.927	0.859	0.95
NPV	0.903	0.9301	0.901	0.819	0.901
MCC	0.905	0.93	0.902	0.82	0.89
FPR	0.0791	0.0794	0.089	0.099	0.09
FNR	0.079	0.0793	0.089	0.099	0.09

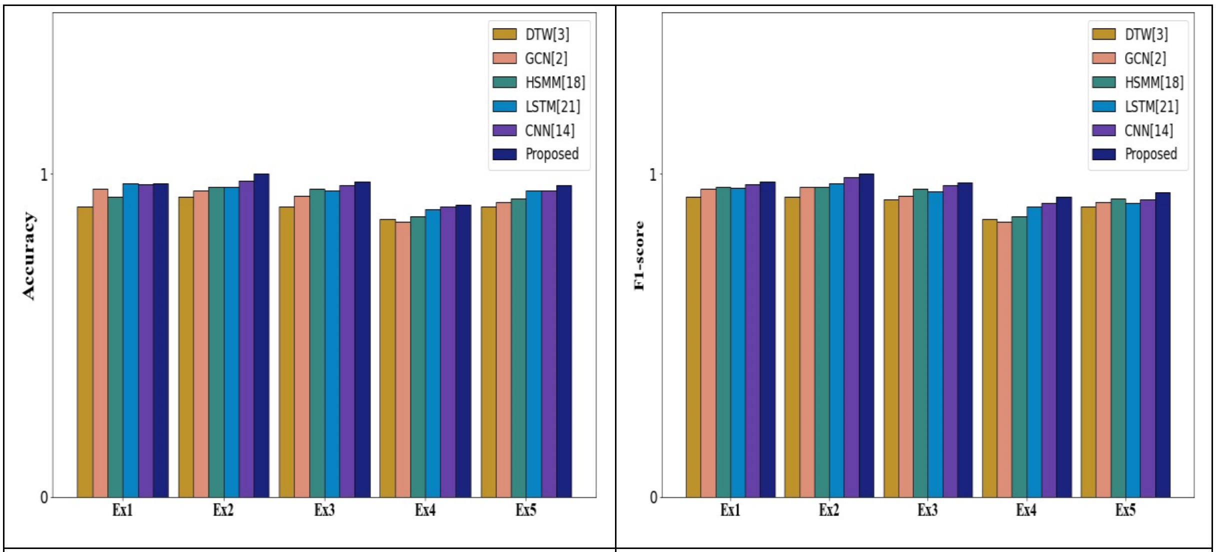
In Table 7, results are shown from an ablation study on a number of feature selection strategies and their impacts on model accuracy. Here, a study was done to analyses how different algorithms perform in five situations, namely Ex1, Ex2, Ex3, Ex4, and Ex5. The method of Chaos Game Optimization (CGO) has shown strong performance. It got 0.9523 in Ex1, 0.9423 in Ex2, 0.9423 in Ex3, 0.9323 in Ex4, and 0.9422 in Ex5 precision evaluations. Ex1 portrays the highest precision as it emphasizes CGO's potential for achieving high levels of accuracy across different environments.

The Aquila Optimizer showed good results. The scores are as follows: 0.9498 in Ex1, 0.9521 in Ex2, 0.9488 in Ex3, 0.9434 in Ex4, and finally 0.9447 in Ex5. One possible reason for the consistently high returns on Ex2 could be that there are better characteristics obtained by this model. However, when compared against 0.9673, 0.962, 0.9689, 0.9593, and 0.9546 accuracies from Ex1 through Ex5, respectively, the Pufferfish Optimization Algorithm performed even better. COA has been recognized for its exceptional consistency in grasping many vital points. This is because Ex3 had the highest precision accuracy among the three. The SOA, or the Serval Optimization Approach, was the most notable one because of the following percentages: 9.9603 in Ex1, 9.9642 in Ex2, 9.9576 in Ex3, 9.9696 in Ex4, and 9.9524 in Ex5. There might be scaling challenges, yet these performances show that MLAs are able to perform well during dimension reduction processes. We have achieved remarkable precision rates utilizing both POA and MLA in the Pufferfish Optimization Algorithm-Serval Optimization Algorithm (POA-SOA), yielding 0.9782 in Ex1, 0.9711 in Ex2, 0.9752 in Ex3, 0.9744 in Ex4, and 0.9726 in Ex5. This demonstrates that combining POA with MLA always gives the highest accuracy among others; hence, Ex1 is the top performer. The feature extraction algorithm adopted greatly influences the accuracy of the model, as the result from this ablation research asserts. While CGO and AO have successfully done this, POA, on the other hand, has demonstrated the highest accuracy, which brings out the significance of picking out an appropriate feature extraction technique under particular circumstances.

Table 7: Ablation Study

Feature Extraction algorithms	**ACCURACY**				
	Ex1	**Ex2**	**Ex3**	**Ex4**	**Ex5**
CGO (Chaos Game Optimization)	0.9523	0.9423	0.9423	0.9323	0.9422
AO (Aquila Optimizer)	0.9498	0.9521	0.9488	0.9434	0.9447
POA (Pufferfish Optimization Algorithm)	0.9673	0.962	0.9689	0.9593	0.9546
SOA (Serval Optimization algorithm)	9.9603	9.9642	9.9576	9.9696	9.9524
POA-SOA (Pufferfish Optimization Algorithm - Serval Optimization algorithm	0.9782	0.9711	0.9752	0.9744	0.9726

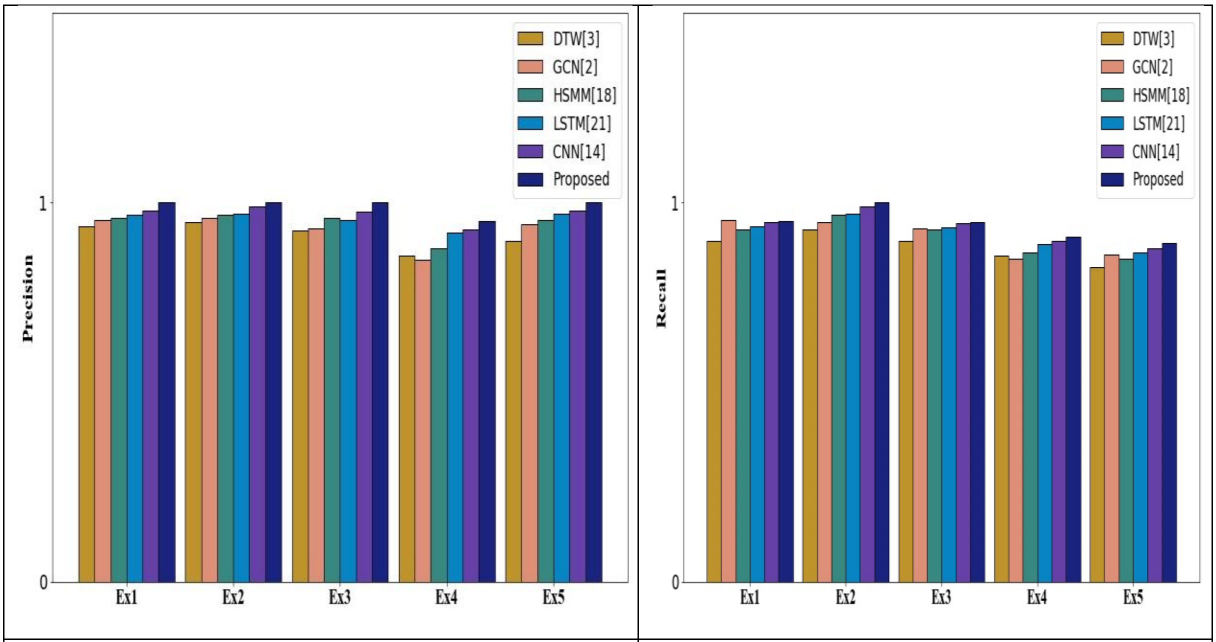
DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024



(a) **(b)**

Figure 10. Graphical format of the performance metrics – (a) Accuracy (b) F1-Score of the models

Figure 10 illustrates graphical performance metrics of the models (a) Accuracy and (b) F1-Score. They offer simplicity and clarity when it comes to assessing how effective the model was. It is clear that the Figure 7(a) and Figure 7(b), Accuracy and F-Score of the proposed model (CPOA) is high during the Ex1, Ex2, Ex3, Ex4, and Ex5 when it is compared to the other model's metrics.

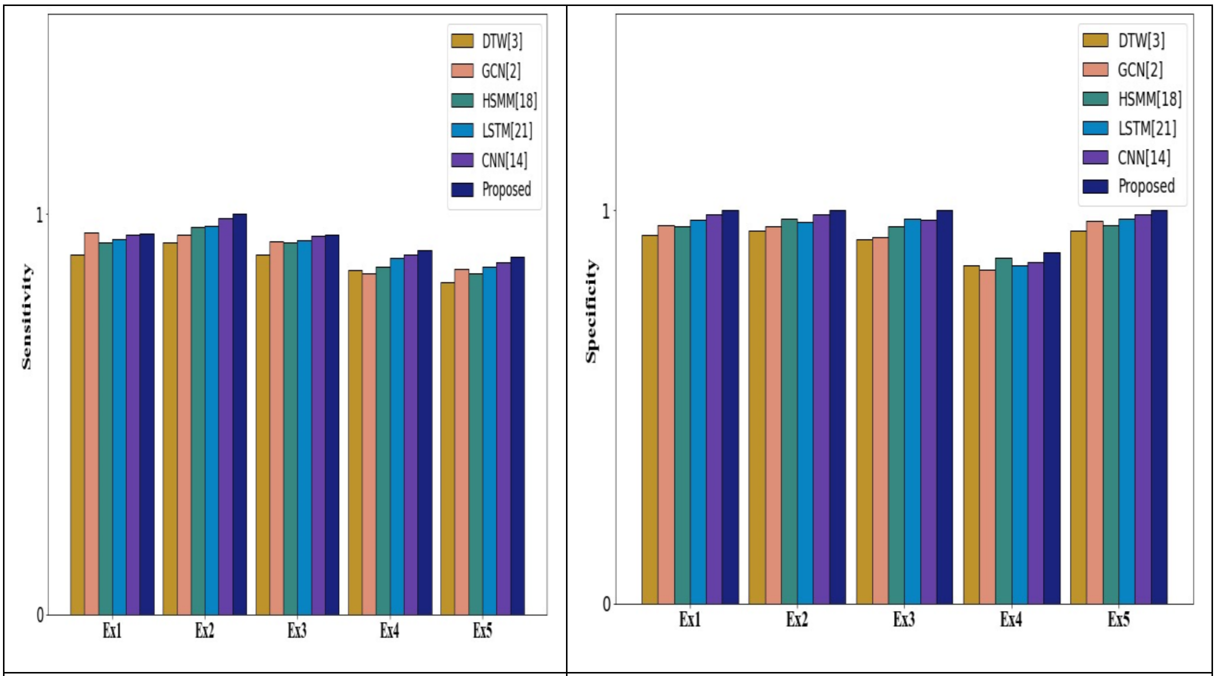


(a) **(b)**

Figure 11. Graphical format of the performance metrics- (a) Precision (b) Recall of the models

In Figure 11, provide a graphical representation of performance metrics for the models. Part (a) showcases Precision, while part (b) illustrates Recall. These visual formats offer a clear and informative way to evaluate the model's precision and ability to retrieve relevant data. It is clear that the Figure 8(a) and Figure 8(b), Precision and recall of the proposed model (CPOA) is high during the Ex1, Ex2, Ex3, Ex4, and Ex5 when it is compared to the other models metrics.

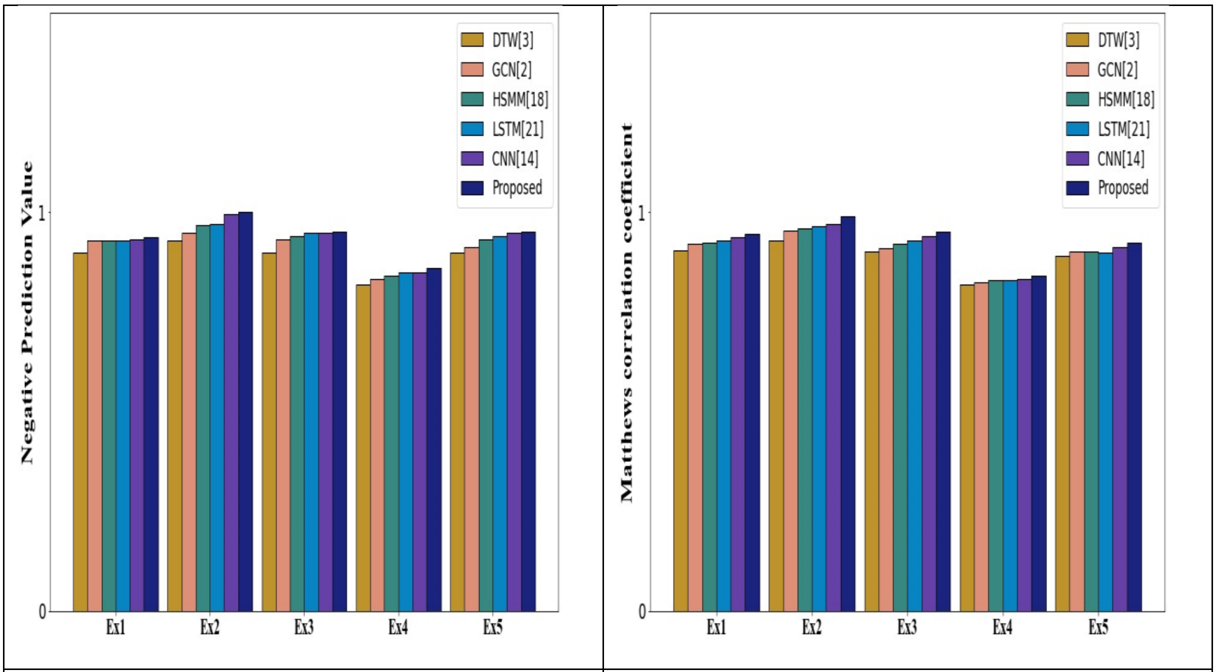
DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024



(a) **(b)**

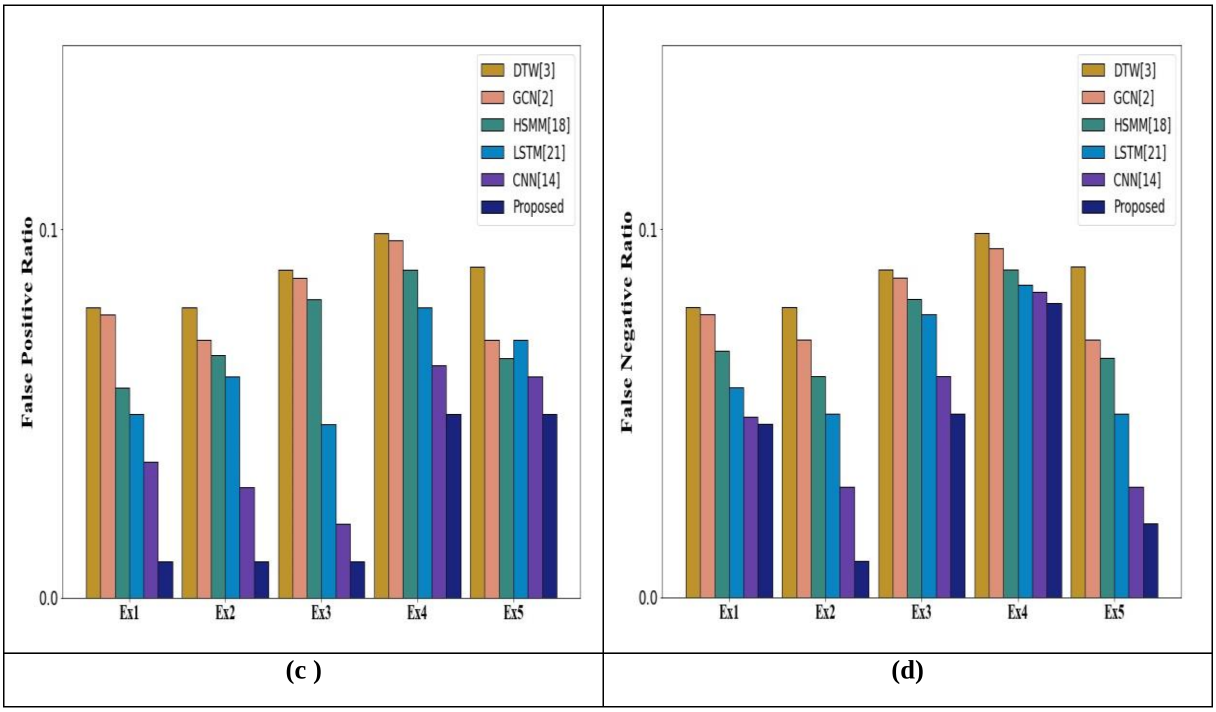
Figure 12. Graphical format of the performance metrics- (a) Sensitivity (b) Specificity of the models

Within Figure 12, a graphical display of performance metrics is presented. Part (a) focuses on Sensitivity, while part (b) emphasizes Specificity. These visual formats provide an insightful means of assessing the model's ability to detect true positives and avoid false positives. It is clear that the Figure 9(a) and Figure 9(b), sensitivity and specificity of the proposed model (CPOA) is high during the Ex1, Ex2, Ex3, Ex4, and Ex5 when it is compared to the other models metrics.



(a) **(b)**

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024



(c) **(d)**

Figure 13. Graphical format of the performance metrics- (a) NPV (b) MCC (c) FPR (d) FNR for the models

Figure 13(a)(b) (c) and (d), illustrates the graphical format of the performance metrics like NPV, MCC, FPR, FNR for the proposed and the other models. It is clear that the Figure 10(a) and Figure 10(b), MCC and NPV of the proposed model -CPOA is high during the Ex1, Ex2, Ex3, Ex4, and Ex5 when it is compared to the other model's metrics. Like that From the Figure 10 (c) and (d) it is clear that the FPR and FNR of the proposed model during the Ex1, Ex2, Ex3, Ex4 and Ex5 is low when it is compared to the DTW[3], GCN[2], CNN[14], LSTM [21] and HSMM[18].

5. Conclusion

In conclusion, research endeavors in the multimodal human movement assessment have yielded promising results and advanced methodologies. Through meticulous image pre-processing, ensuring the consistency of skeleton data joint position images. The utilization of Generative Adversarial Networks (GANs) for data augmentation, coupled with the Contrast Limited Adaptive Histogram Equalization technique, significantly enhanced both dataset size and image quality. Introduced a novel feature extraction framework encompassing Joint-level, Limb-level, Temporal, and Statistical Features, enriching our understanding of the data. The innovative hybrid feature selection model PutterFish Customized Serval Optimizer (PFCSO), combining the Pufferfish Optimization Algorithm (POA) and Serval Optimization algorithm (SOA), demonstrated remarkable efficiency. The development of a RehabNet++architecture, featuring an optimized Multilayer Perceptron, Attention-based Convolutional Neural Networks, Transfer Learning and STR-ResNet architecture, marked a significant advancement in human movement assessment. Fine-tuning MLP's hidden layers using the hybrid POA and SOA optimization model further improved detection accuracy. The integration of Grad-CAM visualization provided valuable insights into model decisions. In comparative analysis, the proposed deep learning architectures displayed superior performance against baseline approaches, confirmed through rigorous statistical testing. Using PYTHON as a tool of choice, this research contributes to the ever-evolving field of multimodal human movement assessment, opening new horizons for accurate and insightful analysis.

Acknowledgement

The author extends the appreciation to the Deanship of Postgraduate Studies and Scientific Research at Majmaah University for funding this research work through the project number (R-2024-1254).

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

References

- **[1]** Amatriain-Fernández, S., Murillo-Rodríguez, E.S., Gronwald, T., Machado, S. and Budde, H., 2020. Benefits of physical activity and physical exercise in the time of pandemic. Psychological Trauma: Theory, Research, Practice, and Policy, 12(S1), p.S264.
- **[2]** Deb, S., Islam, M.F., Rahman, S. and Rahman, S., 2022. Graph convolutional networks for assessment of physical rehabilitation exercises. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 30, pp.410-419.
- **[3]** Yu, X. and Xiong, S., 2019. A dynamic time warping based algorithm to evaluate kinect-enabled home-based physical rehabilitation exercises for older people. Sensors, 19(13), p.2882.
- **[4]** Zhang, W., Su, C. and He, C., 2020. Rehabilitation exercise recognition and evaluation based on smart sensors with deep learning framework. IEEE Access, 8, pp.77561-77571.
- **[5]** Wu, N.N., Tian, H., Chen, P., Wang, D., Ren, J. and Zhang, Y., 2019. Physical exercise and selective autophagy: benefit and risk on cardiovascular health. Cells, 8(11), p.1436.
- **[6]** Semwal, V., Singh, G., Crespo, U. and González, R., 2021. Heterogeneous Computing Model for Post-injury Walking Pattern restoration and Postural Stability Rehabilitation Exercise Recognition.
- **[7]** Capecci, M., Ceravolo, M.G., Ferracuti, F., Iarlori, S., Monteriu, A., Romeo, L. and Verdini, F., 2019. The KIMORE dataset: KInematic assessment of MOvement and clinical scores for remote monitoring of physical REhabilitation. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 27(7), pp.1436-1448.
- **[8]** Baltzer W I, Smithostrin S, Warnock J J, et al. Evaluation of the clinical effects of diet and physical rehabilitation in dogs following tibial plateau leveling osteotomy[J]. Journal of the American Veterinary Medical Association, 2018, 252(6):686-700.
- **[9]** Rochester, C.L., 2019. Patient assessment and selection for pulmonary rehabilitation. Respirology, 24(9), pp.844-853.
- **[10]** Jiang, Y., 2020. Combination of wearable sensors and internet of things and its application in sports rehabilitation. Computer Communications, 150, pp.167-176.
- **[11]** Kitzman, D.W., Whellan, D.J., Duncan, P., Pastva, A.M., Mentz, R.J., Reeves, G.R., Nelson, M.B., Chen, H., Upadhya, B., Reed, S.D. and Espeland, M.A., 2021. Physical rehabilitation for older patients hospitalized for heart failure. New England Journal of Medicine, 385(3), pp.203-216.
- **[12]** Fossati, C., Torre, G., Vasta, S., Giombini, A., Quaranta, F., Papalia, R. and Pigozzi, F., 2021. Physical exercise and mental health: The routes of a reciprocal relation. International Journal of Environmental Research and Public Health, 18(23), p.12364.
- **[13]** Zhu, Z. A., Lu, Y. C., You, C. H., & Chiang, C. K. (2019). Deep learning forsensor-based rehabilitation exercise recognition and evaluation. Sensors, 19(4),887.
- **[14]** Panwar, M., Biswas, D., Bajaj, H., Jöbges, M., Turk, R., Maharatna, K. and Acharyya, A., 2019. Rehab-net: Deep learning framework for arm movement classification using wearable sensors for stroke rehabilitation. IEEE Transactions on Biomedical Engineering, 66(11), pp.3026-3037.
- **[15]** Sun, M., Zhao, S., Gilvary, C., Elemento, O., Zhou, J. and Wang, F., 2020. Graph convolutional networks for computational drug development and discovery. Briefings in bioinformatics, 21(3), pp.919-935.
- **[16]** Liao, Y., Vakanski, A. and Xian, M., 2020. A deep learning framework for assessing physical rehabilitation exercises. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 28(2), pp.468-477.
- **[17]** Esraa Mohamed, The Relationship between Artificial Intelligence and Internet of Things: A quick review, Journal of Cybersecurity and Information Management, Vol. 1 , No. 1 , (2020) : 30-34 (Doi : https://doi.org/10.54216/JCIM.010101)
- **[18]** Capecci, M., Ceravolo, M.G., Ferracuti, F., Iarlori, S., Kyrki, V., Monteriu, A., Romeo, L. and Verdini, F., 2018. A Hidden Semi-Markov Model based approach for rehabilitation exercise assessment. Journal of biomedical informatics, 78, pp.1-11.

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024

[19] Caru, M., Curnier, D., Bousquet, M. and Kern, L., 2020. Evolution of depression during rehabilitation program in patients with cardiovascular diseases. Disability and Rehabilitation, 42(3), pp.378-384.

[20] Mandolesi, L., Polverino, A., Montuori, S., Foti, F., Ferraioli, G., Sorrentino, P. and Sorrentino, G., 2018. Effects of physical exercise on cognitive functioning and wellbeing: biological and psychological benefits. Frontiers in psychology, p.509.

[21] Rahman, Z.U., Ullah, S.I., Salam, A., Rahman, T., Khan, I. and Niazi, B., 2022. Automated detection of rehabilitation exercise by stroke patients using 3-layer CNN-LSTM model. Journal of Healthcare Engineering, 2022.

[22] Schez-Sobrino, S., Vallejo, D., Monekosso, D.N., Glez-Morcillo, C. and Remagnino, P., 2020. A distributed gamified system based on automatic assessment of physical exercises to promote remote physical rehabilitation. IEEE Access, 8, pp.91424-91434.

[23] Iwao, Y., Shigeishi, H., Takahashi, S., Uchida, S., Kawano, S. and Sugiyama, M., 2019. Improvement of physical and oral function in community-dwelling older people after a 3-month long-term care prevention program including physical exercise, oral health instruction, and nutritional guidance. Clinical and Experimental Dental Research, 5(6), pp.611-619.

[24] Whyte, J., Dijkers, M.P., Fasoli, S.E., Ferraro, M., Katz, L.W., Norton, S., Parent, E., Pinto, S.M., Sisto, S.A., Van Stan, J.H. and Wengerd, L., 2021. Recommendations for reporting on rehabilitation interventions. American Journal of Physical Medicine & Rehabilitation, 100(1), pp.5-16.

[25] Aditya Sharma , Aditya Vats , Shiv Shankar Dash , Surinder Kaur, Artificial Intelligence enabled virtual sixth sense application for the disabled, Fusion: Practice and Applications, Vol. 1 , No. 1 , (2020) : 32-39 (Doi : https://doi.org/10.54216/FPA.010104)

[26] Reem Atassi , Fuad Alhosban, Fusion Optimization and Classification Model for Blockchain Assisted Healthcare Environment, Fusion: Practice and Applications, Vol. 9 , No. 2 , (2022) : 62-73 (Doi : https://doi.org/10.54216/FPA.090205)

[27] Khder Alakkari, Alhumaima Ali Subhi, Hussein Alkattan, Ammar Kadi, Artem Malinin, Irina Potoroko, Mostafa Abotaleb, El-Sayed M El-kenawy, Forecasting COVID-19 Infection Using Encoder-Decoder LSTM and Attention LSTM Algorithms, Journal of Intelligent Systems and Internet of Things, Vol. 8 , No. 2 , (2023) : 20-33 (Doi : https://doi.org/10.54216/JISIoT.080202)

[28] Ashish Sharma , Yogesh Sharma , Radhika Bansal , Sushant Verma, Implementation of Crowd Sale using ERC-20 Tokens, Journal of Cybersecurity and Information Management, Vol. 2 , No. 1 , (2020) : 05-12 (Doi : https://doi.org/10.54216/JCIM.020101)

[29] Mustafa Altaee, Talib A., M. A. Jalil, Ali J., Thamer A. Alalwani, Intelligent Multi-Level Feature Fusion Using Remote Sensing and CNN Image Classification Algorithm, Journal of Intelligent Systems and Internet of Things, Vol. 9 , No. 1 , (2023) : 36-48 (Doi : https://doi.org/10.54216/JISIoT.090103)

DOI: https://doi.org/10.54216/FPA.170112
Received: November 25, 2023 Revised: March 18, 2024 Accepted: July 15, 2024