

Rehabilitation Recommendation and Diet Plans for ACL Tear Athletes

Dr. T. Rajasenbagam

Department of CSE, Government college of technology, Coimbatore, Tamil Nadu, India 641013

A. Thanusavarthini ME

Department of CSE, Government college of technology, Coimbatore, India 641013

Abstract - The Anterior Cruciate Ligament (ACL) injury requires proper diagnosis, personalized treatment, and monitoring that will allow for full recovery and return to sports activities. The current paper provides an efficient approach to developing a procedure of AI-assisted rehabilitation process taking into consideration ACL tear classification, personalized diet formulation, and monitoring effectiveness of patients' physical exercises.

Firstly, a Convolutional Neural Network (CNN), enhanced using Improved Human Evolutionary Algorithm (IHEA) was used for classifying injuries associated with ACL tears with high efficiency and accuracy. In case the extent of injuries is estimated, an appropriate treatment method could be developed. As opposed to other approaches that provide recommendations regarding nutrition of ACL injury patients via a set of rules and language models, the paper suggests the usage of GRU-based Variational Autoencoder (GRU-VAE) to generate personalized nutritional programs considering such individual factors of each patient as body mass index, physical activity, and stages of recovery. Finally, in order to monitor the performance of the patient's physical activities, HHDCNN with an integrated architecture has been applied since it analyzes the spatial-temporal aspects of joint motion and measures the quality of movements made in real time. The HHDCNN network with a hybrid architecture is applied in order to evaluate the effectiveness of the exercises performed through detection of spatial and temporal features of joint motion and assessment of the quality of actions performed.

The system developed includes a web-based application that displays joint angles, repetitions, scores, and corrective advice. It was found from the experiments conducted that the developed rehabilitation system has higher effectiveness than currently used systems owing to higher personalization, providing real-time exercise performance information, and use of multimodal approaches of artificial intelligence.

Keywords: Deep learning, HHDCNN, Anterior Cruciate Ligament tear, AI dashboard, exercise recommendations.

1. INTRODUCTION

Torn ACL injuries are among the common knee injuries found in sports people and physically active individuals. One must identify whether the injury is partial or complete to decide what treatment options to follow as well as the type of rehabilitation required for each, which sometimes even entails undergoing surgery. As such, MRI is mainly employed in identifying ACL injuries as it generates high resolution images[1]. Nonetheless, interpretation of the images needs experts' interpretation, and this might result in delays in deciding next steps to take.

The emergence of AI within the health care industry has led to innovations in designing automated ACL injuries and rehabilitation systems. CNNs are the mostly adopted DL networks for image classification due to their good performance in generating results. However, training DL models poses a number of challenges including lack of sufficient number of training images, quality of pictures, existence of noise, and variation in angle when capturing knee pictures. In addition, tuning of model hyper parameters, such as learning rate and filter sizes, is a challenging task.

The system incorporates a user-friendly interface that stores all the fundamental details of patients such as their age, height, weight, injury severity, athlete or non-athlete, and if there are any comorbidities

such as diabetes mellitus. Based on these attributes, the system proposes a personalized dietary regime using a systematic data set, where the selection of food would be carried out in an organized manner considering GI values and nutritional needs. What sets this solution apart from others using rules and large language models is the systematic data set model combined with GRU-VAE[9].

Furthermore, the system provides personalized physical therapy routines according to the level of the injury sustained and monitors their execution in real-time. Using pose estimation techniques and HHDCNN, the system evaluates patient activities, estimates joint angles at hip, knee, and ankle joints, counts repetitions, and provides instant feedback and evaluation scores[2]. Torn ACL is shown in Fig 1.

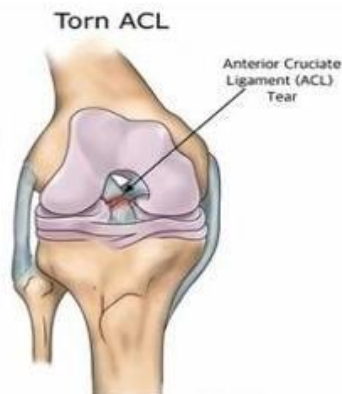


Figure 1 shows the Torn ACL

The key objective of the current study is to develop a comprehensive, intelligent, and patient-centered ACL injury rehabilitation system that integrates medical imaging, nutritional therapy, and exercise prescription. It aims to aid clinical decision making, promote patient engagement, and accelerate the recovery process. The current study serves as a solid foundation for future studies, including wearable sensors, three-dimensional movement analysis, and clinical application on a large scale.

2. RELATED WORKS

2.1 Deep Learning for ACL Tear Classification

Over the past few years, deep learning techniques have been widely used in enhancing the diagnosis of ACL injuries with the help of MRIs. The use of CNNs is quite advantageous since they are capable of automatically discovering essential characteristics from images without any human intervention[1]. Furthermore, there are some investigations that have used the integration of CNNs with optimization algorithms such as IHEA to choose suitable model parameters automatically. It is important to note that this integration has yielded better results in terms of classification of ACL status into normal, partial tear, and total tear.

2.2 Deep Learning for Rehabilitation Exercise Evaluation

As computer vision technology progresses, deep learning becomes a popular approach for analyzing the performance of patients during their rehabilitation exercises. The Hybridized Hierarchical Deep Convolutional Neural Network (HHDCNN) model is one example of an algorithm that can be used for monitoring the performance of such exercises. Through the use of the algorithm, patients' movements can be quantitatively analyzed based on scores[2]. In comparison with other approaches, HHDCNN offers superior results.

2.3 AI-Based Diet Recommendation Systems

Artificial intelligence has also found widespread application in nutrition-based diets to make customized nutritional regimes[9]. The latest approaches use neural networks, including Variational Autoencoders (VAE) together with

language models to recommend meals based on the user's age, body mass index (BMI), and any health problems. Although the above systems may recommend a variety of meals, there is no assurance that the meal suggestions will be consistent and accurate regarding their nutritional content.

2.4 Deep Learning-Based Motion Recognition in Rehabilitation

Knowledge of the mechanics of human motion is crucial for any rehabilitation program[6]. The application of deep learning approaches, such as CNN and recurrent neural networks, has been applied to detect and classify various human motions. Such models can extract the spatial and temporal information of the movement process.

2.5 Sensor-Based and Vision-Based Rehabilitation Monitoring

The most common approaches that are applied for rehabilitation monitoring include those using wearable sensors and camera-based techniques[8]. The former offers an accurate way of measuring but at the cost of high price and low accessibility. The latter technique makes use of cameras and deep learning algorithms for capturing body movements.

2.6 RESEARCH GAP ANALYSIS

However, despite advances in techniques for detecting tears in ACLs, such as use of convolutional neural networks (CNN), GRU-based methods, and optimization approaches, there are still major issues yet to be addressed. First, existing techniques only concern themselves with the detection process from MRI images and use simple 2D modeling or manual analysis, which does not account for possible cases of partial and subtle tears. Second, the amount and availability of existing data sets are low, uneven, and of low quality, making it

impossible for any system to generalize. Third, most systems limit themselves to binary classification, without taking the level of the tear into account.

Secondly, no unified approach is used to tackle all of the problems. Instead, there are separate methods dedicated to detection, recommendation of diet, and exercises. For example, diet recommendations are often personalized and account for various criteria, including glycemic index and medical condition. However, there are few systems able to detect tear and recommend exercises in the same time frame.

In addition, the issue of manual parameter tuning restricts performance and scalability in deep learning. Furthermore, there are not enough platforms that incorporate diagnosis, personalized planning, and monitoring for users. In order to compensate for these shortcomings, this study introduces a combined AI approach for ACL rehabilitation with optimized CNN severity classifier, GRU-VAE based diet plan from datasets, and HHDCNN for real-time exercise monitoring in one platform.

2.7 LIMITATIONS OF EXISTING AI-BASED ACL REHABILITATION SYSTEMS

Although there have been considerable improvements made in artificial intelligence technology in terms of detecting ACL injuries, rehabilitation, and healthcare recommendations, some limitations are currently facing these methods, including:

I. Dependence on Manual Diagnosis:

Current models for ACL detection largely depend on the expertise of radiologists and orthopaedic experts for analysis based on MRI images. This method tends to be highly subjective, hence affecting the accuracy and effectiveness of diagnosis.

II. Limited and Non-Standardized Datasets:

The amount of available MRI data required for categorizing ACL injuries into normal, partial tear, and complete tear is not adequate, nor are they well labeled. In addition, variations in quality, slice thickness, and imaging protocol of the MRI scans affect the performance of the AI models.

III. Incomplete Feature Representation in CNN Models:

Some researchers apply basic CNN architecture in the analysis process or use single-plane MRI images. This approach limits the ability to learn complete features of the ACL due to its complicated anatomy.

IV. Lack of Hyperparameter Optimization:

Several models in deep learning require manual selection of the hyperparameters, which cannot guarantee their efficiency in other datasets. This problem negatively affects the overall efficiency of the model and impedes their scalability in clinical practice.

V. Lack of Integrated Rehabilitation Systems:

Most of today's approaches concentrate on one particular component, such as the diagnosis of the disease or exercise supervision, but not an integration of several components including detection of severity, dietary recommendations, and monitoring of the rehabilitation process.

VI. Absence of Real-Time Feedback Mechanisms:

The lack of the function that enables real-time feedback while performing exercises prevents patients from adopting a correct position and decreases the efficiency of the

process as well as increasing the risks of re-injury.

VII. Limited Personalization in Diet and Rehabilitation Planning:

Today, most models that make dietary recommendations use the rule-based system or generic AI. However, there is no account of the glycemic index, medical history (for instance, diabetes), and activity. Also, most rehabilitation programs are generic rather than individual.

3. PROPOSED METHODOLOGY

The proposed system offers a complete ACL rehabilitation mechanism that is part of the intelligent dashboard as a whole. This system approach provides a complete ACL rehabilitation solution. The flow of the classification is as depicted in fig 3.1

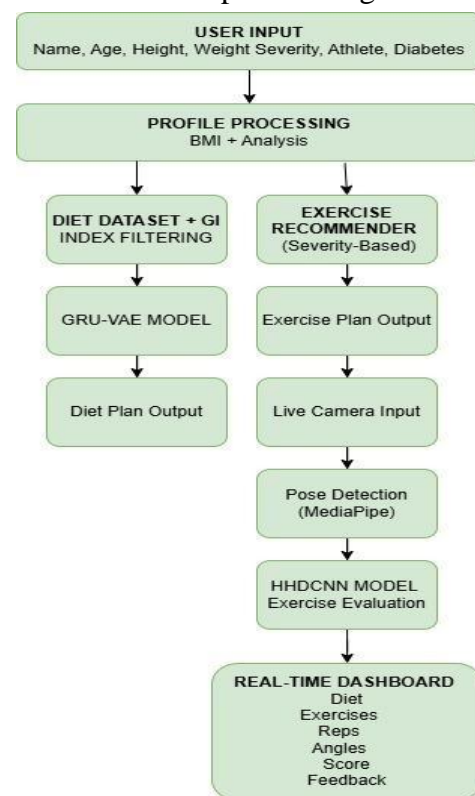


Figure 3.1 Architecture diagram

3.1 DATASET PREPARATION

The suggested rehabilitation framework adopts a joint usage of motion datasets and IFCT datasets to facilitate proper exercise

evaluation, diet planning based on the patient's condition, and rehabilitation programs according to the injury severity. The KIMORE (KINematic assessment of MOVement and REhabilitation) dataset is applied to develop an AI model capable of assessing the quality of exercises. Meanwhile, the exercise and diet datasets function as sources of structured data that can be used to create personal rehab and nutrition plans for patients.

3.1.1 KIMORE DATASET FOR REHABILITATION MONITORING

Regarding the recognition and evaluation of exercise movements, the proposed system mainly uses the KIMORE (KINematic assessment of MOVement and REhabilitation) dataset. It should be noted that this database is specially developed to analyze exercises performed during rehabilitation therapy. Its main features include multiple rehabilitation exercises that are focused on lower limb movements, the use of RGB cameras, depth sensors, and motion capture techniques, kinematics information at a joint level, including knee angles and hip movements, and body posture alignment. Thus, this dataset can become a perfect source for developing a model, which requires both spatial and temporal motion characteristics. Joint level movements are shown in figure 3.2

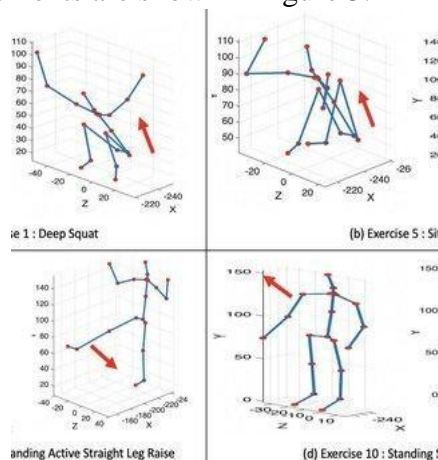


Figure 3.2 Joint Movement in X, Y, Z axis

3.1.2 EXERCISE DATASET FOR REHABILITATION PLANNING

Along with the use of KIMORE system, an ACL rehabilitation exercise dataset has been adopted to suggest the exercise routine according to the severity of the ACL injury. The dataset includes the following variables; Exercise Name, Rehabilitation Phase (Early, Mid, Advance), Number of sets and repetitions, Guidelines to execute, Muscles worked (quadriceps, hamstrings, knee). Such a dataset allows developing a step-by-step rehabilitation technique and decreases the chances of suggesting incorrect exercise.

3.1.3 DIET DATASET FOR PERSONALIZED NUTRITION PLANNING

A well-organized Indian Food Composition Table is used for developing diet plan for every patient. In contrast with AI-based diet plan generation systems, such a dataset-driven solution helps in making consistent and clinically reliable suggestions.

The dataset contains Meal categories (Breakfast, Lunch, Snack, Dinner), Food item, Nutrition value (Calories, Proteins, Carbohydrates, Fats, Glycemic Index).

3.2 HYBRIDIZED HIERARCHICAL DEEP CONVOLUTIONAL NEURAL NETWORK (HHDCNN)

HHDCNN initialization has been carried out to increase the accuracy of motion capture, image segmentation of sports athletics exercise rehabilitation. Multi-layered Convolutional Network Neural Architecture will be used, with Fully Connected Layer and Readout Layer selection being made. The figure below presents the proposed HHDCNN Architecture.

The main reason for making this selection is the locality nature of features of such a

framework. This makes it possible for the network to consider and retain its temporal input data in the form of multi-channel synchronized time sequences. It only considers relations and does not relate any data in the input signal from regions temporarily apart from each other. Figure 3.3 demonstrates the proposed Hierarchical Hybridized Convolutional Neural Network architecture.

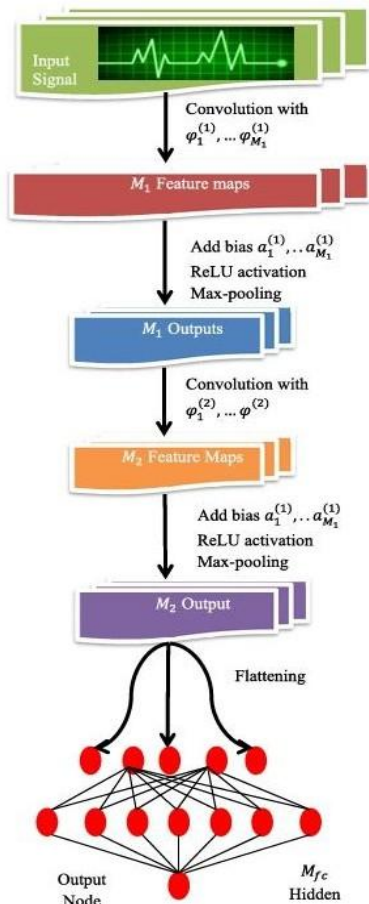


Figure 3.5 The proposed hybridized hierarchical convolutional neural network architecture.

In the first layer of the convolution, $y_{j,i}$ feature maps are produced through convolution with M_1 kernels denoted as $\phi_{j,i}^{(1)}$ of size K_1 and $i \in [1, M_0]$. The processed data in terms of inertial sensor information, which is in form of $j=1, \dots, K_0$ samples and $i=1, \dots, M_0$ channels, will be fed into the network as inputs. Bias term denoted as $a_i^{(1)}$ will be added to each feature map prior to

generating the output of the first convolutional layer through an activation of Rectified Linear Unit as indicated in equation (1).

$$b_l^{(1)} = \text{ReLU} \left(\sum_i \phi_{i,l}^{(1)} * y_i + a_l^{(1)} \right) \quad (1)$$

After that, all the layers of convolution can have max pooling layers to help in downsampling of the feature maps through imposing on the maximum values in non-overlapping lengths windows d . downsample factor of $1/d$. At first, output of the max-pooling layer is referred to as $q_{j,i}^{(1)}$ for $j \in [1, \frac{K_0}{d}]$, $i \in [1, M_1]$.

This process will repeat for the next convolution layer with different kernel size and bias. For instance, let us consider the input data as $q_{j,i}^{(2)}$ and form the feature map with a convolution operation on $l=1, \dots, M_2$ kernels of size $\phi_{j,i,l}^{(2)}$ of size $j=1, \dots, K_2$ and $I \in [1, M_1]$. Then the bias $a_l^{(2)}$ is added to obtain the output value after applying the Rectified Linear Unit Activation Function.

$$b_l^{(2)} = \text{ReLU} \left(\sum_i \phi_{i,l}^{(2)} * q_{i}^{(2)} + a_l^{(2)} \right) \quad (2)$$

As shown in equation (2), where $l \in [1, M_2]$. After that, the output $q_{j,i}^{(2)}$ which has a size of $j \in [1, K_0/d_2]$ and $i \in [1, M_2]$ is subjected to max pooling. Likewise, this process continues till $\log_d K_0$ convolutional layers and max pooling are weighted followed by downsampled of the input data to one.

The output of the last max-pooling layer goes through the fully connected layer $q_{j,i}^{(2)}$ which becomes $\tilde{q}_{j,i}^{(2)}$ of size $j \in [1, M_2 K_0/d_2]$ vector. The weight of this vector is $S_{j,i}^{(fc)}$ with $j=1, \dots, (M_2 K_0/d_2)$ and $i=1, \dots, M_{fc}$. Further adding the bias term, $a^{(fc)}_i$ and passing through the Rectified Linear Unit activation function provides the fully connected layer output value with $I \in [1, M_{fc}]$.

$$q_i^{(fc)} = \text{ReLU} \left(\sum_j S_{j,i}^{(fc)} \tilde{q}_{j,i}^{(2)} + a^{(fc)}_i \right) \quad (3)$$

Lastly, an output layer is used to reduce the M_{fc} hidden layer output nodes to the particular target parameter motion frame. This is done through multiplication with the weight vector $S^{(ro)}_{j,1}$ with $j \in [1, M_{fc}]$. A bias value of $a^{(ro)}$ is added to achieve the evaluation of the target parameter.

$$x = \sum_j S^{(ro)}_{j,1} q^{(fc)}_j + a^{(ro)} \quad (4)$$

Likewise, a target variable (x_1, \dots, x_m) that describes the motion features in a set can be obtained by adjusting the number of dimensionality of bias and weights in the layer. The choice of the threshold is given below.

$$R = r[y, x, f(y, x), p(y, x)] \quad (5)$$

From Equation (5), where $p(y, x)$ is the local feature elements in the region around the change point while $f(y, x)$ is the gray value at point (y, x) of the picture. R is a function of the point (y, x) and $p(y, x)$. Feature description of the super-pixel is made, and a process of training is performed using the softmax classifier. Loss function is given by,

$$I(\theta) = - \frac{1}{n} \left[\sum_{j=1}^{\{n\}} \sum_{i=1}^{\{l\}} k\{x^{(j)}\} = \right. \\ \left. \frac{\sum_{i=1}^{\{l\}} \{e^{\theta_i y^{(j)}}\}}{\sum_{k=1}^{\{l\}} \{e^{\theta_k y^{(j)}}\}} \right] + \\ \sum_{j=1}^{\{\lambda\}} \sum_{i=1}^{\{m\}} \theta_{ji}^2 \quad (6)$$

In the context of (6), θ_i refers to the vector weight which has to undergo training before acquisition, and m is the corresponding dimension. In this case, parameter l refers to the class in all the samples in the training set while n refers to the number of training samples. If $x(j)=i$, then the result will be one, otherwise, zero. To avoid overlapping, a certain value must be added, and λ , in this regard, refers to the normalization constant. $Kx(j)=i$ represents an indicator function. It is possible to find the weight vector after

training. The probability distribution of superpixel W_j where $j \in \{1, \dots, t\}$ can be determined by hw_j, o

$\phi_p(y_j, y_i)$ refers to the color and pixels of the potential energy of paired. The mathematical formulation is expressed as,

$$\phi_p y_j, y_i = \mu(y_j, y_i) \sum_{n=1}^l s^{(n)} I^{(n)}(f_j, f_i) \quad (7)$$

This is shown in equation (7) where $\mu(y_j, y_i)$ a label is an inter changeability function. The term $\mu(y_j, y_i) = (y_j \neq y_i)$ is used to deal with the situation where the same adjacent pixels are marked with the label. The symbol $s^{(n)}$ is a Gaussian kernel function weight while $I^{(n)}(f_j, f_i)$ is a Gaussian kernel, which is used. The parameters j and i represent the location in the feature space. The symbols f_j and f_i are the feature vectors of the pixel. The Gaussian kernel is presented as,

$$l(f_j, f_i) = s^{(1)} e^{-\frac{|p_j - p_i|^2}{2\theta_\beta^2} - \frac{|J_j - J_i|^2}{2\theta_\alpha^2}} + \\ s^{(2)} e^{-\frac{|p_j - p_i|^2}{2\theta_\beta^2}} \quad (8)$$

As shown from equation (8), the first kernel uses the color and location data of the pixel. The symbols p_j and p_i represent the exact locations of the pixels j and i whereas J_j and J_i are the color values of pixels j and i respectively. The symbols θ_β and θ_α show the similarity between the color and pixels respectively. In the kernel function that follows, the use of the location data of the pixels is done only to provide location data. The algorithm transforms the message processing step into a function space step. It helps to reduce the complexity of messages from second order to first order.

Even with regard to the reverse reconstruction stage, the Gibbs sampling algorithm is applied. However, for the problem of inconsistency arising in the dimension due to the pooling function, the zero function is carried out in the process of

de-convolution. This visible layer consists of real value units compared to the previous step of forward inference:

$$U_{\beta}^r \sim M \left[\sum_r D_{\beta}^r \otimes \text{rot} (L_{d,r}, 180^{\circ}) + c_d, \gamma^2 \right] \quad (9)$$

Equation (9) represents $G(\varepsilon)$, which is an evaluation measure that needs to be minimized. In this case, $G(\varepsilon)$ can be defined as $\text{rmsq}(\varepsilon)$, representing RMSE of the mini-batch.

Deep learning algorithms have been employed because of the presence of computing capabilities. Threshold Segmentation Algorithm has been selected for its accurate segmentation in images and classification of motion types in order to avoid injury from inappropriate postures. Classification and efficiency measures of data sets include several kinds of validation techniques.

3.3 VARIATIONAL AUTO ENCODER

In the framework of the diet recommendation system, the Variational Autoencoder (VAE) serves as one of the crucial components. The VAE algorithm comprises two stages – the encoder and the decoder. The architecture of the VAE is depicted in Figure 3.4

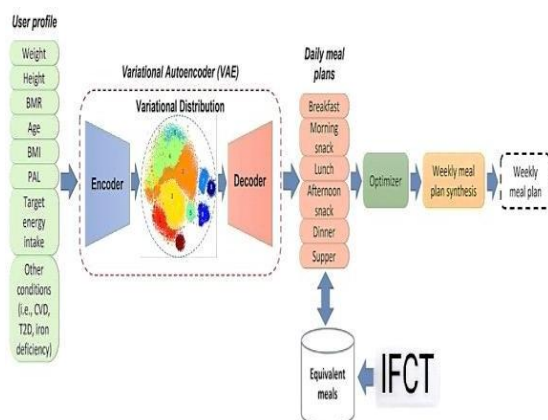


Figure 3.4 Overview of the proposed diet recommendation method.

Basically, the Variational Autoencoder takes the user's profile as the input, which includes, but not limited to, his/her age, current body mass index, and specific health problems. However, instead of using these features directly, the VAE maps the input data to the latent space, where they will be represented in a more meaningful way, including all critical characteristics required for a particular diet. One of the major strengths of the latent space is that it can reveal some patterns in the data structure. For instance, those users that have similar health conditions or need the same nutrition are closer to each other. This allows for generating personalized diets for each particular person, since similar people are considered together. Moreover, using the latent space is helpful from the perspective of explainability of the system. Since users having similar characteristics are located close to each other in the space, the system is able to justify the generated recommendation in the following manner: similar users obtain similar diets.

I. Encoder: Learning User Representation

It is critical to consider the role of an encoder when trying to make sense of the user. The encoder must receive the user profile (age, weight, height, gender, health state, severity, physical activity, among others) and encode this data into something that will be more meaningful and compact. Let us call the user profile: $x = (x_1, x_2, \dots, x_u) \in \mathbb{R}^u$

The input vector is transformed by the fully connected layer to the following feature representation: $x'=f(x)$. The purpose of this step is to make sure that the information in the feature vector can reveal meaningful insights about the user data.

- BMI Calculation:

$$BMI = \frac{Weight}{\left(\frac{Height}{100}\right)^2}$$

- BMR Calculation:

$$BMR = (10 \times Weight) + (6.25 \times Height) - (5 \times Age) + C$$

Where:

$C = +5$ for males / $C = -161$ for females

- Activity Adjustment:
 Calories = BMR \times Activity Factor

If Athlete $\rightarrow 1.5$ and Non-athlete $\rightarrow 1.3$

- Diabetes-Aware Filtering: low Glycemic Index ($GI < 55$)

II. Latent Space Representation

Rather than use the feature vector as is, the model maps the feature vector into the latent space. This means that the encoder must produce two vectors:

- Mean vector: $\mu = f_{\mu}(x'), \mu \in \mathbb{R}^d$
- Standard deviation vector:
 $\log(\sigma^2) = f_{\sigma}(x'), \sigma \in \mathbb{R}^d$

These define a multivariate Gaussian distribution: $\mathcal{N}(\mu, \sigma)$

III. Reparameterization Trick

To allow training through backpropagation, the model uses the reparameterization trick

$$z = \mu + \sigma \odot \epsilon$$

where:

- $\epsilon \sim \mathcal{N}(0,1)$ (random noise)
- z is the sampled latent vector

IV. Decoder: Generating Meal Plans

After we have obtained the latent vector z , the decoder must transform it into a meal plan. This means that we need to use GRUs since the meal plans follow an order (breakfast, snacks, lunch, and dinner).

V. GRU-Based Meal Sequence Generation

The decoder generates **5 meals per day** as breakfast, snack, lunch, dinner, immunity boosters.

The GRU operates as follows:

$$h(t) = \begin{cases} \text{GRU}(z), & t = 1 \\ \text{GRU}(h(t-1)), & t > 1 \end{cases}$$

where:

- $h(t)$ = hidden state at time step t

VI. Energy and Nutrient Prediction

In addition to meal types, the model also predicts:

- Total Energy Intake (Calories)
 $EI = \sum_{t=1}^T f_{EI}(h(t))$
- Nutritional Values (Macronutrients)
 $\hat{n} = \sum_{t=1}^T f_{nutr}(h(t))$

where:

- f_{EI} : predicts calories
- f_{nutr} : predicts nutrients (protein, carbs, fats, etc.)

VII. Loss Functions

To ensure the system generates **healthy and accurate meal plans**, multiple loss functions are used:

(a) Macronutrient Loss ensures the nutrients stay within recommended ranges:

$$L_{macro} = \frac{1}{N} \sum_{i=1}^N (| \min_val(i) - n(i) | + | \max_val(i) - \hat{n}(i) |)$$

(b) Target Energy Calculation

$$EI = \begin{cases} BMR \cdot PAL + D, & BMI \leq 18.5 \\ BMR \cdot PAL, & 18.5 < BMI < 25 \\ BMR \cdot PAL - D, & BMI \geq 25 \end{cases}$$

- Underweight \rightarrow increase calories

- Normal → maintain
- Overweight → reduce

4. EXPERIMENTAL RESULTS AND CONCLUSION

The essential breakthrough in the system is the Pose Estimation that utilizes MediaPipe Pose for landmark detection such as Hip, Knee, Ankle, and Foot. From Feature Extraction of landmarks, the system calculates knee Angle, Hip Angle, and Ankle Angle Symmetry between both legs. Figure 4.1 shows the proposed outcome.

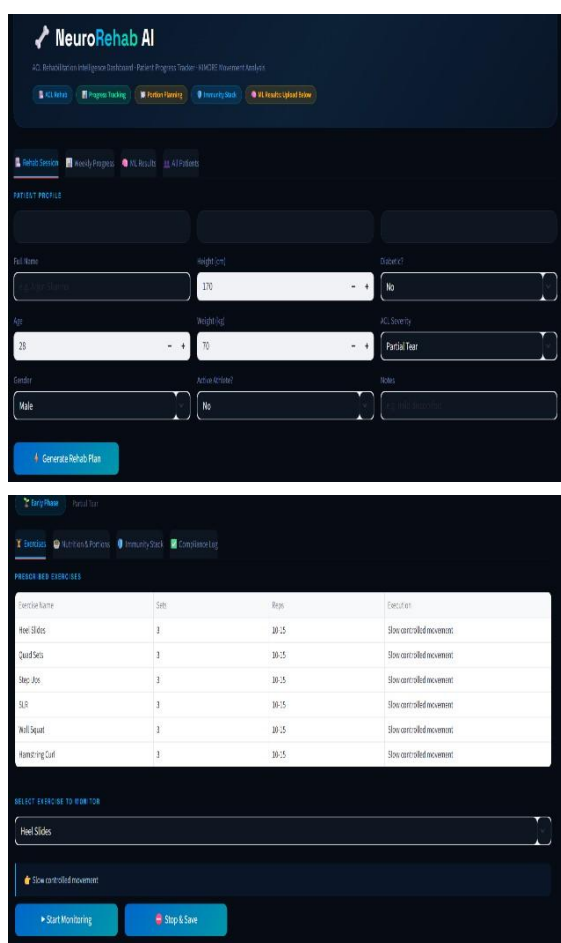


Figure 4.1 Interactive dashboard

The Movement Analysis involves the repetition calculation using threshold angle values, motion smoothing through temporal filtering, and sequence analysis of joint motion. After feature extraction, the data are then processed in the HHDCNN Model. The HHDCNN Model incorporates spatial

and temporal learning, determines exercise correctness, and categorizes movement quality. Lastly, the system calculates a form score (range 0-100) based on joint angle accuracy, movement stability, symmetry, depth, and posture.

4.1.1 Movement Velocity (Control of Motion)

A velocity-based diagram helps understand how consistently the patient performs movements. Patients with limited capabilities have an irregular and unpredictable pace of movement. High-scoring subjects have a consistent pace of movements. It is associated with increased neuromuscular coordination, thus allowing minimizing excessive stress of the knee joint.

4.1.2 Mean Knee Angle (Postural Consistency)

It can be seen from the data provided that the high-scored patients perform consistent and correct movements of joints. However, a scatter plot of values in low-scored patients is indicative of wrong posture and poor body coordination. Thus, it is proved that the device is able to estimate the quality of movements, not only their existence.

4.1.3 Hip Joint Coordination

The hip joint maintains stability of lower limb movement. The observed clustering of values in high-scored patients shows that patients have correct hip joint placement and, thus, minimize knee joint overload.

4.2 PERFORMANCE COMPARISON WITH EXISTING METHODS

The comparison graph drawn using the reference study suggests that performance increases with the increment in data sample size. The suggested method obtains an accuracy rate of 98.7% at a dataset size of 1900 samples, which outperforms all other

existing methodologies. Such superior results are achieved owing to incorporating biomechanics, the combination of learning models, and improved representation of movement behavior. Not like the existing methodologies, the suggested system not only recognizes the gesture visually but also identifies how it should be performed. Table 1 demonstrates the performance ratio evaluation of all models.

Model	Train Acc	Val Acc
HHDCNN (Tang 2020)	~94%	~89%
LightGBM	99.16%	91.07%
XGBoost	93.89%	89.18%
Random Forest	95.56%	90.58%
Stacking Ensemble	—	91.66%

Table 1. The performance ratio

4.3 PRECISION-RECALL ANALYSIS

The precision-recall graphs help understand the capability of the model to differentiate between the various rehabilitation classes (low, mid, and high). The precision-recall graphs obtained from your models are located quite close to the ideal space,

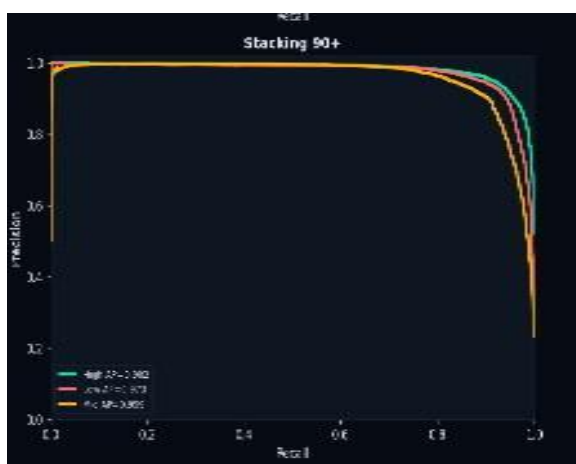


Figure 4.2 Precision Recall analysis

suggesting almost negligible false detection and very effective classification of each

level. Figure 4.2 shows the Precision recall analysis.

Out of all the tested models, the stacking-based methodology has achieved the highest results in terms of precision at all possible recall values.

4.4 ROC CURVE ANALYSIS

The ROC curves also further authenticate the efficacy of the proposed system. All systems show good performance with stacked classifiers showing best separability among all models. All curves are near the top left corner, which shows great classification performance by the model. It indicates that the proposed system will be able to distinguish between poor recovery, average recovery, and perfect recovery with high confidence. Figure 4.3 shows the Roc curve analysis.

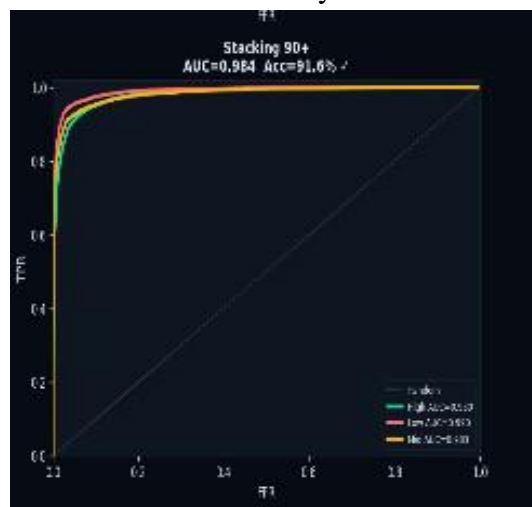


Figure 4.3 ROC Curve

4.5 CONFUSION MATRIX ANALYSIS

The confusion matrix gives a detailed analysis of classification effectiveness for different rehabilitation classes. The output indicates that the low class, which represents poor recovery, is the most accurately classified while the mid class, representing average recovery, also shows

high classification rate with some confusion caused due to recovery overlap. The high class representing perfect recovery also has shown good performance in terms of accurate classification. Figure 4.4 shows the confusion matrix of HHDCNN.

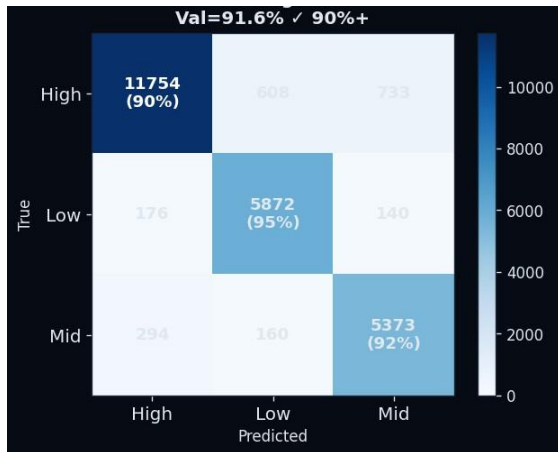


Figure 4.4 Confusion matrix of Stacking Ensemble

5. OVERALL SYSTEM INTERPRETATION

In this regard, the above-mentioned system integrates successfully the principles of biomechanics (joint angles, symmetry, and motions), machine learning algorithms (Random Forest, XGBoost, and LightGBM), and ensemble learning approach (stacking). This means that it is capable not only to recognize various movements and their patterns but to evaluate the quality of these movements in terms of their clinical significance. As can be seen from the analysis of the entire set of charts and obtained results, the suggested system provides accurate evaluation of the rehab process, detects all the essential biomechanical features, outperforms conventional HHDCNN-based systems, and allows recommending an adequate diet for patients.

REFERENCES

1. Nantong- Stomatological Hospital, "Anterior Cruciate Ligament Tear Detection Based on Combination of Convolutional Neural Network Enhanced by Improved Human Evolutionary Algorithm"(2024).

2. D. Tang, "Hybridized hierarchical deep convolutional neural network for sports rehabilitation exercises," IEEE Access, vol. 8, pp. 118969–118977, Jul. 2020, doi: 10.1109/ACCESS.2020.3005189.
3. Y. Su, "Implementation and rehabilitation application of sports medical deep learning model driven by big data," IEEE Access, vol. 7, pp. 156338–156348, 2019.
4. J. Lian and G. Hui, "Human evolutionary optimization algorithm," Expert Syst. Appl., vol. 241, May 2024, Art. no. 122638.
5. K. Joshi and K. Suganthi, "Anterior cruciate ligament tear detection based on convolutional neural network and generative adversarial neural net work," Neural Comput. Appl., vol. 36, no. 9, pp. 5021–5030, Mar. 2024.
6. Y.Liao, A.Vakanski, and M.Xian, "A deep learning framework for assessing physical rehabilitation exercises," IEEE Trans. Neural Syst. Rehabil. Eng., vol. 28, no. 2, pp. 468–477, Feb. 2020.
7. M. Kulkarni and R. A. Hamid Khan, "Anterior cruciate ligament tear detection: ML and DL approaches," in Proc. 4th Int. Conf. Sustain. Expert Syst. (ICSSES), Oct. 2024, pp. 1091–1094.
8. M. J. Awan, M. S. M. Rahim, N. Salim, A. Rehman, and B. Garcia-Zapirain, "Automated knee MR images segmentation of anterior cruciate ligament tears," Sensors, vol. 22, no. 4, p. 1552, Feb. 2022.
9. Ilias Papastratis , Dimitrios Konstantinidis , Petros Daras & Kosmas Dimitropoulos, "AI nutrition recommendation using a deep generative model and ChatGPT" Scientific Reports | (2024) 14:14620 | <https://doi.org/10.1038/s41598-024-65438-x>.
10. Csanalosi, M. et al. Personalized nutrition for healthy living (protein-study): Evaluation of a mobile application in subjects with type 2 diabetes and prediabetes. Diabetologie und Stoffwechsel (2023).
11. Marsall, M., Engelmann, G., Teufel, M. & Bäuerle, A. Exploring the applicability of general dietary recommendations for people affected by obesity. Nutrients 15, 1604 (2023).
12. Shandilya, R., Sharma, S. & Wong, J. Mature-food: food recommender system for mandatory feature choices a system for enabling digital health. Int. J. Inf. Manag. Data Insights 2, 100090 (2022).
13. Niszczota, P. & Rybicka, I. The credibility of dietary advice formulated by chatgpt: robo-diets for people with food allergies. Nutri tion 112, 112076 (2023).
14. EFSA. Scientific opinion on dietary reference values for protein. EFSA J. 10, 2557 (2012).
15. EFSA. Scientific opinion on dietary reference values for carbohydrates and dietary fibre. EFSA J. 8, 1462 (2010).

16. Stefanidis, K. et al. Protein ai advisor: A knowledge-based recommendation framework using expert-validated meals for healthy diets. *Nutrients* 14, 4435 (2022).

17. Toledo, R. Y., Alzahrani, A. A. & Martinez, L. A food recommender system considering nutritional information and user preferences. *IEEE Access* 7, 96695–96711 (2019).

18. Brown, T. et al. Language models are few-shot learners. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. & Lin, H. (eds.) *Advances in Neural Information Processing Systems*, vol. 33, 1877–1901 (Curran Associates, Inc., 2020).