A Machine Learning-Based System for Real-Time Hand Rehabilitation

Abstract - Patients who are currently experiencing a fracture, nerve injury, undergoing physiotherapy, elderly people prone to have reduced hand functionality are dependent on hand rehabilitation. Traditional rehabilitation methods tend to be expensive, time consuming and provide no real time feedback limiting effectiveness and accessibility. The paper presents the development of a more accessible and cost-effective solution to hand rehabilitation. Existing methods either involve expensive equipment or they're far from providing precise real time feedback so that patients cannot exercise appropriately with supervision. To this end, we propose a machine learning based system that provides real time feedback for eight hand exercising using a laptop with webcam. We use the MediaPipe Hand Tracking for detecting hand movements and a Random Forest Classifier for classifying exercises. The model is trained on a self-collected dataset, and each exercise comprises approximately 1260 instances; the hand landmarks are detected using MediaPipe. Evaluation of the accuracy of the presented model showed its high efficiency in detecting hand exercises with 98,61% accuracy. This system has the advantage of being less costly and convenient to adopt, more so it brings about a tremendous improvement in the rehabilitation of patients and also the frequency of patient monitoring in clinical practice.

Index Terms- Hand exercises, MediaPipe, Random Forest classifier, Hand rehabilitation, Hand exercise recognition, Real-time feedback, Rehabilitation technology.

I. INTRODUCTION

Hand rehabilitation is important for patients to regain hand function by recovering from nerve injuries, fractures, post-surgical conditions, or age-related hand impairment [1]. One of the drawbacks of traditional physiotherapy is that it is usually provided by professional one on one sessions that take a good amount of time, can be quite expensive, and isn't convenient in the face of other commitments such as work [2]. This means that many patients are encouraged to continue with their exercises at home, making it hard to maintain proper technique, as well as how to most effective perform them, limiting their progress.

To handle these challenges, several technology-based solutions have been developed including Virtual Reality (VR) systems [3] and Orthosis devices for hand rehabilitation [4]. However, these tools have their own limitations, particularly high costs and complex sensor-based systems [5]. For more delicate hand movements these devices can be difficult to use or deliver accurate, real-time feedback. Moreover, VR systems are uncomfortable to use for long periods of time, without the explicit and detailed corrective feedback required for effective rehabilitation.

Despite recent innovations in hand rehabilitation have focused on improving overall hand function without making more user friendly and affordable solutions available for the hands. Existing technologies are mostly targeted towards body or larger limb movements for rehabilitation, leaving those suffering from, for example carpal tunnel syndrome [6] or post hand surgery without suitable tools for rehabilitation. Furthermore, there are important gaps in systems [7] that able to provide real time feedback necessary to maintaining proper exercise form as well as accelerating recovery.

This research therefore tries to develop a machine learning based model for hand rehabilitation to fill that gap. In order to do this, we present an exercise model comprised of eight specifically chosen exercises that involve finger, palm and wrist movements which will receive real time feedback through a simple setup consisting of laptop and webcam. In comparison to either traditional orthosis device or VR system, this solution is affordable, yet highly effective solution that enables patients to exercise using this solution in a correct manner, with comfort, leading to faster recovery process.

Due to the lack of pervasive and affordable hand rehabilitation devices [8], the technology presented in this study aims to offer precise, timely feedback at a user friendly, and inexpensive price. This model provides a practical means of making targeted hand rehabilitation more available and could embed physiotherapy practices and improve patient outcome.

The primary contributions of this paper are summarized as follow:

- the development of a comprehensive hand exercise detection system that integrates data collection, model design, and real-time feedback to enhance hand rehabilitation.
- By utilizing a webcam for data capture and employing a Random Forest Classifier along with MediaPipe for accurate hand gesture recognition, the system delivers precise and adaptive feedback during specific hand exercises.
- The proposed model not only ensures accurate detection and classification of hand movements but also provides users with real-time correction and offering a practical and cost-effective solution for improving rehabilitation outcomes.
- This approach addresses the limitations of existing hand rehabilitation technologies by focusing on ease of use, affordability, and effective performance in detecting fine motor movements.

The rest of the paper is organized as follows: Section **II** presents an overview of the related work. Section **III** details the methodology, including data collection, preprocessing, model development, and feedback mechanisms. Section **IV** reports the experimental results. Section **V** include discussion of the research and finally, Section **VI** summarize the paper with conclusion and future direction of this research project.

II. RELATED WORK

In this section, we will talk about the recent trends in hand tracking technology and hand rehabilitation technology, putting a spotlight on how MediaPipe an open-source framework developed by Google can be used to help. In the first instance, we will present general usages of MediaPipe across various domains like augmented reality, healthcare, education, and exercise detection. We then will look at how MediaPipe's hand tracking tech has been specifically used for hand rehabilitation exercises comparing it to traditional hand rehabilitation systems and devices, and possibly what it might do to change hand rehab forever.

2.1 MediaPipe

Google's MediaPipe has taken off as a fast-growing open source framework for building real-time machine learning pipelines across a wide variety of applications [9]. Since its launch, MediaPipe has advanced rapidly and employed versatile solutions in its application in healthcare [10], augmented reality [11], entertainment [12], and education [13]. The multi stage pipeline is used in this modular, cross platform framework to process and analyze video in real time which makes this a very robust tool for tasks where power efficiency at the lowest latency is needed.

MediaPipe is extensively used in creating interactive filters, 3D effects, and virtual objects in augmented reality (AR) [14]. For example, apps like Snapchat and Instagram depend heavily on frameworks such as MediaPipe to film ways of overlaying effects on facial landmarks and hands to improve the experience [15] , and also this is being used in gaming [16] and virtual meetings [17] where facial expression and movement are animated in real time.MediaPipe is used to power real time action recognition and body pose estimation [18] in the domain of video analytics and is particularly useful in the areas of security surveillance, sports performance analysis, and entertainment. For instance, media pipe is being used by sports scientists as a means of body pose estimation on athletes to simply identify an athlete's form [19] enabling immediate feedback to improve their techniques and avoid injury. In the area of fitness applications for example, MediaPipe can tell users where they are in a workout and if it is being performed correctly [20].

MediaPipe is also used for healthcare. MediaPipe has been used in the broader field of medicine to extract movement of a human, such as detecting posture problems or movement disorders [21]. Telemedicine, it can be incorporated into systems that monitor facial expressions to detect psychopathology or understand patient movements to assess the degree of physical rehabilitation [22]. Moreover, its use for the tracking of body parts in prosthetics and assistive devices, where accurate real time tracking is essential [23], presents new opportunities for more interactive and responsive healthcare solutions. In another work [24], real time hand movement is recognize with a computer connected camera and applied to support educational activities for young children. The study employed MediaPipe technology to use virtual interactions as augmented control of a mouse for exploring nature and other learning experiences indoors that are generally limited due to the COVID-19 pandemic. This approach also implemented aspects of Korea’s revised Nuri curriculum focused around physical exercise, health and nature exploration.

2.2 Hand Tracking and Exercise Detection with MediaPipe

Among MediaPipe's many capabilities, one of its most useful tools for real time applications is its hand tracking system. MediaPipe hand tracking pipeline will detect and track 21 landmarks on hand in 3D space and give us information about the movements of the hand and the position of finger. This level of precision is precisely what is needed for applications that require fine motor executions such as virtual keyboard interactions, gesture-based controls and rehabilitation exercises.

This capability to tracking hands and fingers with such high level of accuracy has been utilized across a variety of applications, such as interactive gaming [25] , virtual reality [26] and healthcare [27]. In gaming, the MediaPipe helps to make gaming more immersive since you use the hand gestures to control in game actions. Hand tracking in Virtual Reality allows you to interact with environment using natural actions: grabbing items, powering devices, swiping, interacting with tools.

One of the most potent applications of MediaPipe’s hand tracking technology is when it comes to Health driven applications, namely rehabilitation [28]. In particular, hand rehabilitation for patients recovering from surgeries, fractures or nerve injuries necessitates fine and repeated movement of the fingers, palm and wrist. Lastly, MediaPipe allows us to track these movements in real time and get feedback as to whether you did the exercises correctly. One good example is when, for example, a patient needs to do a finger flexion or some wrist extension, the system can detect if the patient does not perform a proper movement and tell the patient to do the movement again.

By integrating with some of the more common tools, such as OpenCV, another computer vision library, MediaPipe has made its tool even more useful for wider applications. OpenCV provides a rich set of low-level functions for image processing, while MediaPipe builds on top of these capabilities to create more high-level abstractions [29]. This enables the solutions to complex vision problems, such as hand exercise detection, with ease. Using MediaPipe's pre trained models for hand tracking and gesture recognition allows the developer to concentrate on the creation of applications that feature specific functionalities [30], like real time feedback in rehabilitation exercises.

2.3 Hand Rehabilitation Systems and Devices

Helping people recover their hand function from conditions or injuries will often involve hand rehabilitation. This process often involves some targeted exercises to strengthen the fingers, palms, and wrists to help patients regain dexterity and mobility [31]. Specific hand exercises are needed to improve muscle strength and use of motor skills for conditions like carpal tunnel syndrome, nerve damage, fracture or after post operative recovery. Traditional rehabilitation systems sometimes are limited by cost and accessibility [32].

In past, hand rehabilitations were dependent on the person sitting in front of professional physiotherapist and are set to undergo physiotherapist sessions. In these sessions, exercises of the hand are guided by therapist as they seek to restore functionality to specific areas of the hand. In person therapy is effective [33], however it is also time consuming, expensive, and [sometimes] difficult for patients to be frequent patients. In addition, when patients are asked to perform exercises independently at home, there is no guidance or supervision, and the exercise is done improperly or produces lower results. The concern is that tools will need to be developed that will bridge this gap and give the patients the real time feedback they need in order to complete the exercises themselves.

Countless problems associated with these challenges have been tackled by numerous technological solutions, especially hand rehabilitation devices. In fact, these devices are traditionally simple and low-tech devices like grip strengtheners to high tech responsive gloves and sensor based systems. For example, robotic gloves [34] help patients do range of motion practice with specific factors of resistance or constraint that are produced by the motorized components. Nevertheless, they are often too expensive and difficult to employ, making them inaccessible to most except specialized rehabilitation centers [35]. Such systems, either sensor based (using motion capture technology or pressure sensitive gloves) offer detailed feedback but are impractical for everyday home use [36].

Additional methods exploring how virtual reality (VR) systems may deliver immersive hand rehabilitation exercises have also been explored. The systems allow patients to interact with a virtual environment performing real world tasks requiring fine motor skills [37]. Although VR systems can be highly motivating, they often cannot provide the fine real time feedback required to correct small scale finger exercises, in particular. Additionally, this type of setup tends to be costly and specialized with the use of expensive hardware, such as VR headsets and motion controllers, which further restricts their use in home-based rehabilitation.

However, the MediaPipe approach provides a low cost and inexpensive alternative using a standard webcam and computer vision and machine learning to track hand movement. In this way, it no longer needs the expensive or bulky hardware, and it becomes a more practical solution for home use [38]. Patients can do their rehabilitation exercises on a laptop or mobile device with a camera, and the system can track movements in real time, displaying instant feedback. That's important because we want to make sure that when we are doing exercises the way they're supposed to be done and the way we are supposed to do them, we are not going to do it with a bad movement, or a movement that doesn't lead to recovery.

The hand tracking system of MediaPipe is also perfectly designed to be of use for hand rehabilitation. It's an excellent tool for exercises that concentrate on hand strength and dexterity as it is able to detect fine motor movements, for example flexion and extension of individual fingers. Real time feedback that users receive is critical to the efficacy of rehabilitation as it allows the user to modify their movements as they go and know that they are following the regime as prescribed. With systems like MediaPipe it is likely that the future of hand rehabilitation systems will involve an increasing integration of machine learning and computer vision technology [39]. Through its affordability and accessibility, MediaPipe can take hand rehabilitation to a whole new level, allowing patients who need frequent, supervised exercises but cannot head into in person physiotherapy sessions. Furthermore, as MediaPipe develops further, its use in hand rehabilitation is likely to further expand with applications of more precise, more interactive exercise systems. MediaPipe afford an exciting new way to build accurate and real-time feedback for hand exercises, overcoming the limitations of traditional approaches to hand rehabilitation devices and methods. This new technology may help make hand rehabilitation more available, less expensive and therefore more effective for patients recovering from hand injuries or conditions.

III. METHODOLOGY

The current portion of the hand exercise detection system involves complete approach, starting from the collection of data using a webcam, followed by data preprocessing, model development, feature extraction, and feedback mechanisms. This section details the steps taken to gather and clean the dataset, the architectural design of the model using MediaPipe and a Random Forest Classifier, and the feedback system created to provide real-time guidance during hand exercises. The overall aim is to ensure that the system captures accurate data, recognizes hand exercises effectively, and offers customized feedback to improve user performance.

3.1 Data Gathering and Hand Exercises, Cleaning

In this hand exercise detection system, the dataset was developed entirely by the author using real-time video capture from a webcam. To create a diverse set of hand movements, the author has performed a series of unique hand exercises, ensuring that the data was collected under varied lighting and surrounding conditions. The eight exercises performed are:

1) Ball Grip (Wrist Down),

2) Ball Grip (Wrist Up),

3) Pinch,

4) Thumb Extend,

5) Opposition,

6) Extend out,

7) Finger bend and

8) Side Squeezer.

All the exercises were recorded roughly for the same amount of time, allowing for a balanced number of images for each exercise. Each exercise was captured from various angles to cover a wide range of scenarios. The webcam was placed at a fixed distance of approximately 50 cm from the ground to minimize fluctuations in distance, thereby reducing potential measurement errors in the collected data.

A. Normal Hand Input

The hand image is captured using a webcam, as shown in the initial step of the process. The webcam position and setup are optimized to maintain a consistent distance and minimize external disturbances.

B. Hand Landmarks Annotation Using MediaPipe

After capturing the hand image, MediaPipe annotates the hand landmarks, identifying 21 key points on the hand. This is visually represented by the annotated hand image in the diagram. The landmarks are crucial for the next step, as they form the basis for feature extraction.

C. Feature Extraction and Saving Landmarks to CSV

The 21 annotated landmarks are converted into 229 inter-coordinates and angles, serving as features for the model. These values are stored in a CSV file for each recorded exercise. This step involves computing distances and angles between landmarks to form a comprehensive feature set.

D. Web Interface for Data Gathering

The web interface, as shown in the image, allows for structured data collection. It includes a timer to ensure that each exercise is recorded for the same duration, maintaining consistency across all exercise recordings.

E. Storing All Coordinates in a CSV File

All the extracted coordinates from each frame, along with their respective exercise labels (e.g., Ball Grip, Thumb Extend), are stored in a CSV file (hand_landmarks.csv). This file contains all the data required for training the model, with entries for each recorded frame and its associated features.

Data Quality and Cleaning

During the video capture process, frames were recorded at 30 frames per second (fps) to avoid excessive data collection. Initially, frames were stored in raw form, and a preliminary review was conducted to assess data quality. After recording each exercise, the number of frames captured and hand landmark coordinates were evaluated for outliers or noisy data. If deviations were found, the process was repeated for that specific exercise to ensure accurate data.

- Quality Review**:** Frames with poor visibility or excessive motion blur were discarded. Around 10-15% of the data was filtered out to remove irregular observations.
- Re-Recording**:** If significant improper images were found, the entire data gathering process was repeated for the affected exercise.
- Labeling and Storing**:** The cleaned frames were labeled according to the specific exercises and stored as feature vectors in the CSV file (hand_landmarks.csv). For training and evaluation, the dataset was split into an 80-20 ratio.

Preprocessing

- The preprocessing involved extracting hand landmarks using MediaPipe, which provided the (x, y) coordinates of 21 key points on the hand. These landmarks formed the primary feature set.
- No resizing, rotation, or flipping of images was performed, as the variations captured during the data gathering were deemed sufficient for training the model.

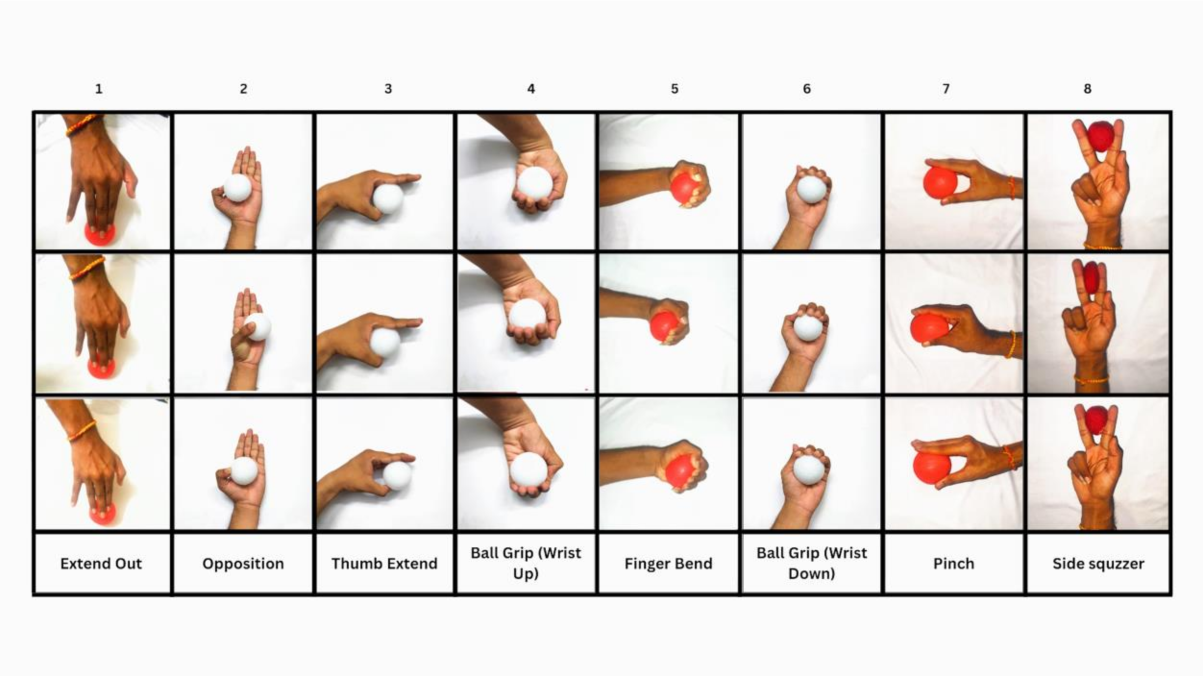


Figure 3.1 Sample Images of dataset

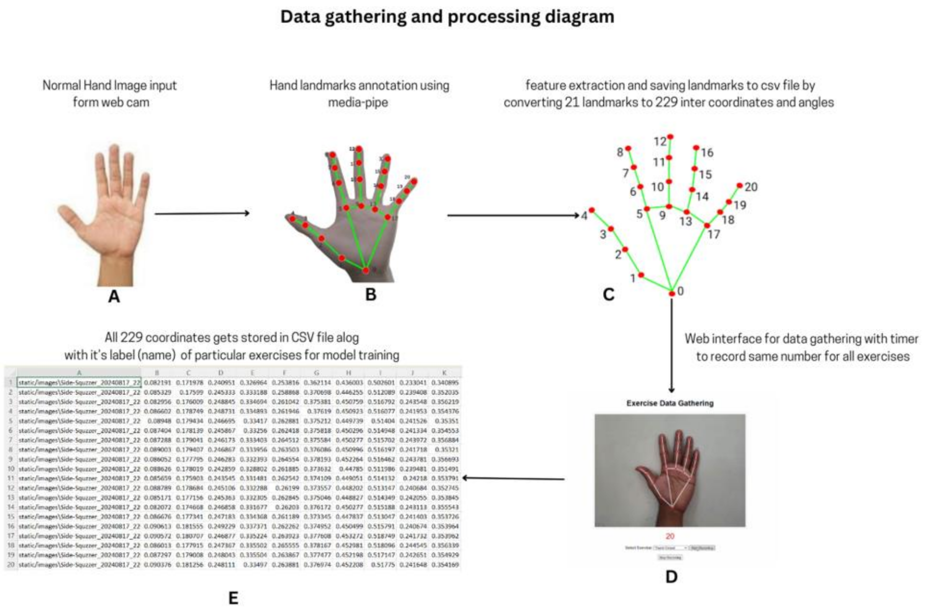


Figure 3.2 Flow diagram of data gathering and pre-processing

3.2 Model Architecture

The hand exercise detection system integrates real-time data preprocessing, feature extraction and machine learning models to give detailed information of the exercises. MediaPipe is used to perform tracking of hand actions, Random Forest Classifier is embedded to recognize signs, while the feedback composition has certainly been developed to give user instructions for properly performing exercises by particular hand movements and key joints of fingers.

A. Real-Time Data Capture

To record hand exercises, the system uses a webcam that captures video frames at a rate of 30 frames per second. This ensures a balance between data sufficiency and computational efficiency. The MediaPipe Hand Landmark Detection model identifies 21 key hand landmarks in each frame, mapping these points accurately over the hand to provide precise information on the positioning and movement of fingers and joints. This forms the foundation for feature extraction.

B. MediaPipe Landmarks Detection

Once the hand is detected, MediaPipe generates a set of 21 hand landmarks representing key points like knuckles, fingertips, and wrist. These landmarks are crucial for tracking hand dynamics and form the basis for calculating distances and angles. The detected landmarks help the system in understanding the spatial arrangement and orientation of the hand in real time.

C. Feature Extraction

Feature extraction is a critical step in the system, where distances between landmarks and angles at the joints are computed. This is done using vector operations on the 2D coordinates of the landmarks to create meaningful features that describe the hand's spatial properties. The key components of feature extraction are:

- Inter-Landmark Distances: The Euclidean distance formula is applied to calculate distances between all landmark pairs. This allows the system to monitor variations in finger spacing and hand posture. The formula used is:

$$\text{distance} = \sqrt{(x^2 - x^1)^2 + (y^2 - y^1)^2}$$

where $(x_1, y_1)$and (x_2, y_2) are the coordinates of two landmarks.

- Angle Between Joints: To determine the angle formed by three consecutive landmarks (e.g., finger joints), the system calculates the angle using the dot product and vector magnitudes:

$$\text{angle} = \arccos\left(\frac{(p1 - p2) \cdot (p3 - p2)}{\| p1 - p2 \| \times \| p3 - p2 \|}\right)$$

This function computes the angles in degrees, which helps the model understand the directional movement and orientation of the fingers during different exercises.

Pseudo Code for Feature Extraction:

```
Function calculate_angle(p1, p2, p3):
    vector_a = p1 - p2
    vector_b = p3 - p2
    dot_product = dot(vector_a, vector_b)
    norm_a = magnitude(vector_a)
    norm_b = magnitude(vector_b)
    angle = arccos(dot_product / (norm_a * norm_b))
    return angle in degrees
Function extract_features(landmarks):
    For each landmark i in landmarks:
        For each landmark j from i+1 to end of landmarks:
            distance = Euclidean_distance(landmarks[i], landmarks[j])
            Append distance to features
    For each landmark i from 0 to len(landmarks) - 3:
        angle = calculate_angle(landmarks[i], landmarks[i+1], landmarks[i+2])
        Append angle to features
    return features
```

These extracted features form a 229-element feature vector, comprising 210 inter-landmark distances and 19 angles, which serve as the input for the classification model.

Saving Features to CSV:

Each frame's landmark features are saved into a CSV file. Each row contains the coordinates of the hand landmarks, the calculated distances, angles, and the exercise label corresponding to the frame. This structured data allows for efficient training and analysis.

Classification Using Random Forest Model:

The feature vectors obtained from the previous step are fed into a Random Forest Classifier from the scikit-learn library. The choice of the Random Forest model is due to its ability to:

- Reduce Overfitting: By aggregating predictions from multiple decision trees, the model reduces the risk of overfitting, making it resilient to noise and imbalanced data.

- Handle Non-Linear Relationships: Hand exercises often involve intricate joint interactions. The Random Forest's ensemble approach captures these non-linear patterns more effectively than traditional linear models.

The model is trained on a balanced dataset to recognize eight distinct exercises: Ball Grip (Wrist Down), Ball Grip (Wrist Up), Pinch, Thumb Extend, Opposition, Extend Out, Finger Bend, and Side Squeezer.

D. Correction and Feedback

After classification, the system evaluates the hand posture and movement against predefined thresholds for each exercise. If a deviation is detected (e.g., a distance exceeding the threshold), feedback is generated in real-time to guide the user in adjusting their posture. The feedback mechanism is critical for ensuring correct exercise form and minimizing the risk of improper hand movements.

Annotated Output:

The annotated image output shows the real-time overlay of detected landmarks, predictions, and feedback directly on the live webcam feed. This visual representation helps users understand the state of their hand exercises instantly, ensuring an interactive and instructive experience.

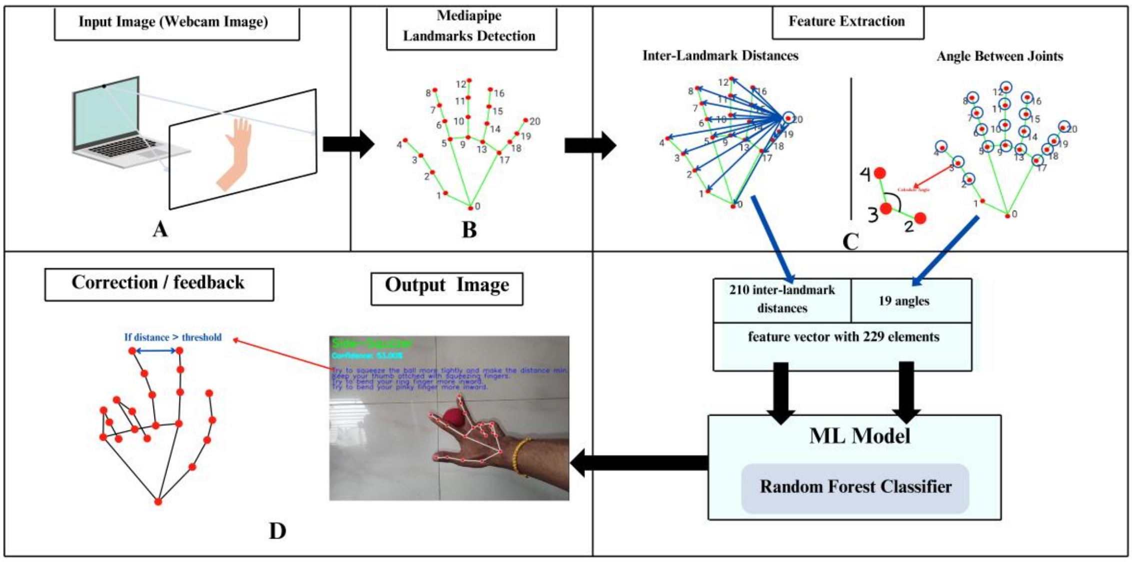


Figure 3.3 Flow diagram of detection and feedback model.

3.3 Feedback Mechanism

The feedback mechanism is a critical component of the hand exercise detection system, providing dynamic and immediate guidance to users during their exercises. By utilizing hand landmark detection and classification, the system ensures that users perform each exercise with correct form, enhancing the overall effectiveness and safety of the workout. The process is streamlined into four key steps as shown in the diagram:

Tailored Feedback for all exercises

For each exercise, a customized feedback function was developed, focusing on the relative positions and orientation of the hand and fingers. For example, in the " Ball Grip (Wrist Down)" exercise, feedback was given based on the distances between the fingertips and MCP joints, as well as the position of the thumb in relation to the index and middle fingers.

Example feedback for the " Ball Grip (Wrist Down)" exercise included:

- Grip Strength: If the distances between the fingertips and MCP joints exceeded a set threshold, the user was advised to adjust their grip.
- Thumb Position: Advice was offered according to the position of the thumb with regards to the other fingers.
- Finger Positioning: The system also checked the alignment of neighboring fingers to ensure proper contact and positioning during the grip.

For each exercise, a similar custom-built feedback system was developed to ensure that users receive meaningful feedback for all specific needs and requirements. Both quantitative and qualitative checks, along with appropriate timings depending on the exercise type, were set in place to ensure correct performance without consuming too much time. The key feedback mechanisms used for each exercise are as follows:

- Ball Grip (Wrist Down): Monitors grip tightness, thumb alignment, and finger spacing.
- Ball Grip (Wrist Up): Focuses on hand orientation and finger curl for a proper upward grip.
- Pinch: Ensures optimal positioning of the thumb and index finger, crucial for an effective pinch.
- Thumb Extend: Provides feedback on thumb extension and finger separation.
- Opposition: Focuses on the thumb's ability to touch the tips of the other fingers.
- Extend Out: Checks the full extension of the fingers and proper spacing.
- Finger Bend: Monitors the bend angles of each finger for correct posture.
- Side Squeezer: Focuses on the grip strength and alignment of the thumb with the rest of the fingers during the squeezing motion.

A. Capturing Input Image: The system captures input images in real-time from a webcam feed during the user’s exercise. This live data serves as the input for hand landmark detection, ensuring that the feedback provided is based on the user’s current posture and movements. The frame rate of 30 frames per second allows for smooth tracking and immediate response.

B. Landmark Detection and Feature Transformation: Upon capturing an image, the MediaPipe model detects 21 hand landmarks. These landmarks are then transformed into a 229-element feature vector, matching the format of the training data stored in the CSV file. This transformation includes calculating distances and angles between landmarks to create a comprehensive representation of the hand's posture and movements. These features are essential for accurately classifying the exercise being performed.

C. Prediction and Exercise Identification: The calculated features are passed through the Random Forest Classifier, which predicts the exercise being performed by matching the input vector against the training data patterns. The classifier outputs the predicted exercise label and its corresponding confidence level, which are then displayed on the live webcam feed. This ensures that users receive immediate feedback on the detected exercise type.

D. Customized Feedback Generation: Based on the predicted exercise, the system provides tailored feedback directly on the webcam feed. The feedback mechanism evaluates the relative positions of the main hand joints and distances between key points (e.g., fingertips, MCP joints) to check if the user is following the correct form.

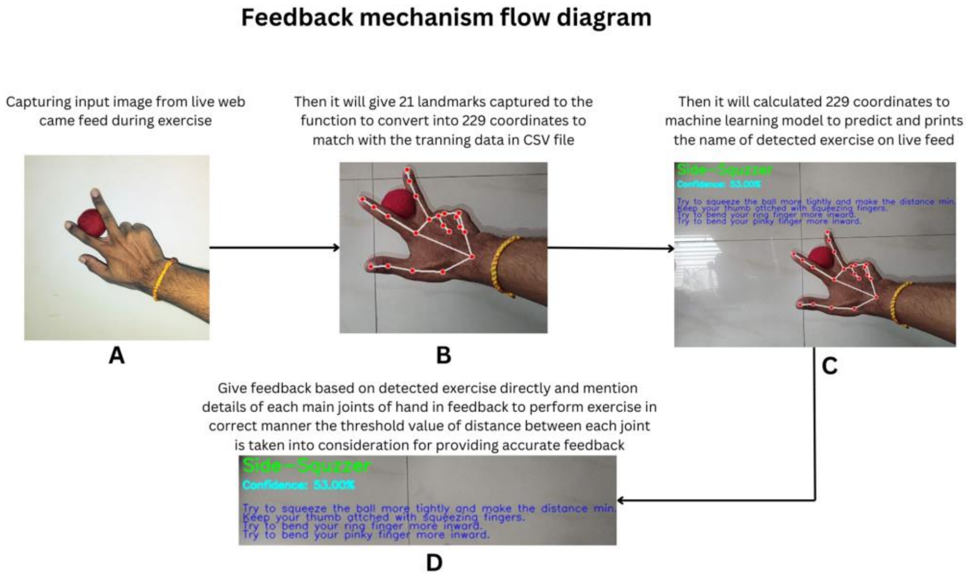


Figure 3.4 Flow diagram of feedback mechanism.

IV. EXPERIMENTAL RESULTS

This section presents the results of our hand exercise detection and feedback system, implemented using a Random Forest Classifier. To ensure optimal model performance, we adjusted the classifier with a set of predefined hyperparameters, which are detailed in Table 4.2. We used Python as the primary programming platform to handle data manipulation and model training. Essential libraries such as pandas, NumPy, MediaPipe, and OpenCV were employed for tasks like data loading, feature extraction, and visualization. The experiments were conducted on a machine with limited RAM and no dedicated GPU, highlighting the efficiency and lightweight computational requirements of the model during both training and usage.

4.1.1. Experimental Setup

The experiments were carried out using a standard laptop configuration without a dedicated GPU, highlighting the system's efficiency even on resource-constrained environments. The webcam setup was maintained at a fixed distance (50 cm from the floor) to ensure consistency in data collection and feedback accuracy.

4.1.2. Software Libraries

The development and implementation leveraged several software libraries:

- Python: The main platform for coding and model implementation.

- Pandas & NumPy: Used for data manipulation and handling.
- MediaPipe: Employed for detecting hand landmarks in real time.
- OpenCV: Utilized for video capture, frame processing, and visualization.

4.1.3. Dataset Description

The dataset used for training and testing the model consists of images of hand exercises captured via a webcam. The dataset is structured as follows:

- Source of Dataset: Self-collected and labelled based on MediaPipe hand landmarks. The collection was conducted under controlled conditions, ensuring consistency in hand positions and angles for each exercise type.
- No. of Classes and Instance Details: The dataset includes 8 different hand exercise classes with an average of 1260 instances per class. Each instance is represented as a labelled vector of hand landmark distances and angles, extracted using MediaPipe.

Table 4.1 Class distribution of the dataset.

Exercise	No. of Images
Side-Squzzer	1583
Extend-Out	1287
Pinch	1282
Thumb-Extend	1272
Opposition	1225
Finger-Bend	1165
Ball-Grip-Wrist-Down	1161
Ball-Grip-Wrist-Up	1077

4.1.4. Hyperparameter Values

To optimize the model, we experimented with different hyperparameter settings. Table 4.2 provides a summary of the hyperparameters used in our final implementation.

Table 4.2 Hyperparameters for the hand exercise recognition system.

Parameters	Values
Model	Random Forest
Number of Trees	100
Batch Size	16
Random state for splitting data	42
Random state for classifier model	22
Feature Scaling	StandardScaler
Test Size	20%

After tuning, the Random Forest model achieved an accuracy of 98.61% on the test set. Table 3 further illustrates the performance of the model under different hyperparameter configurations, demonstrating how variations in the number of estimators and test size affect accuracy.

Table 4.3 Performance of the Random Forest model with varying hyperparameters.

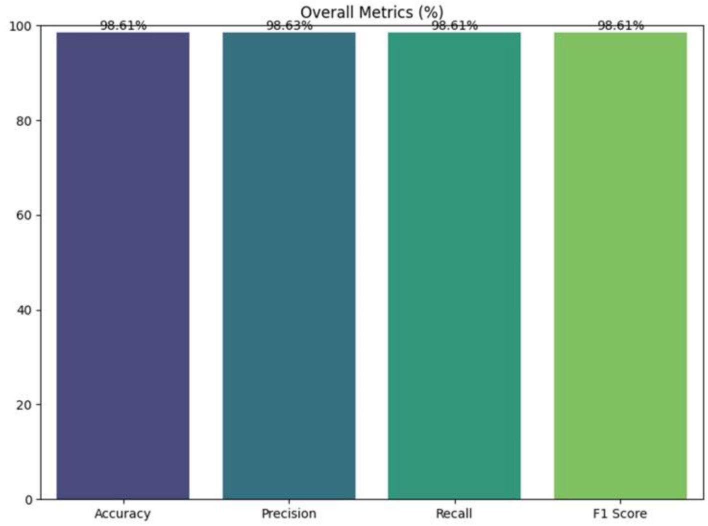
Number of Estimators	Test Size	Accuracy	Random State for splitting	Random State for RF classifier
100	20%	98.61%	42	22
100	25%	98.29%	42	42
50	30%	98.06%	42	42
50	20%	98.22%	42	22

4.1.5. Evaluation Measures

The hand exercise detection model demonstrates high performance across various evaluation metrics, indicating its reliability and accuracy. Below is a detailed analysis based on the obtained results:

1. Model Accuracy: The overall accuracy of the model is 98.61%. This means that out of all the predictions made, 98.61% were correct. The high accuracy suggests that the model is effective at differentiating between the various hand exercises.
2. Precision, Recall, and F1 Score:
    - Precision: The model achieved a precision score of 98.63%. Precision indicates the proportion of true positive predictions out of all positive predictions made by the model. A high precision value suggests that when the model predicts a specific hand exercise, it is highly likely to be correct.
    - Recall: The recall score stands at 98.61%. Recall measures the ability of the model to identify all relevant instances (i.e., all exercises performed). This shows that the model is effective in detecting most of the hand exercises.
    - F1 Score: The F1 score, which is the harmonic mean of precision and recall, is 98.61%. This balanced metric highlights the model's consistency in making accurate predictions while minimizing both false positives and false negatives.
3. Macro and Weighted Averages:

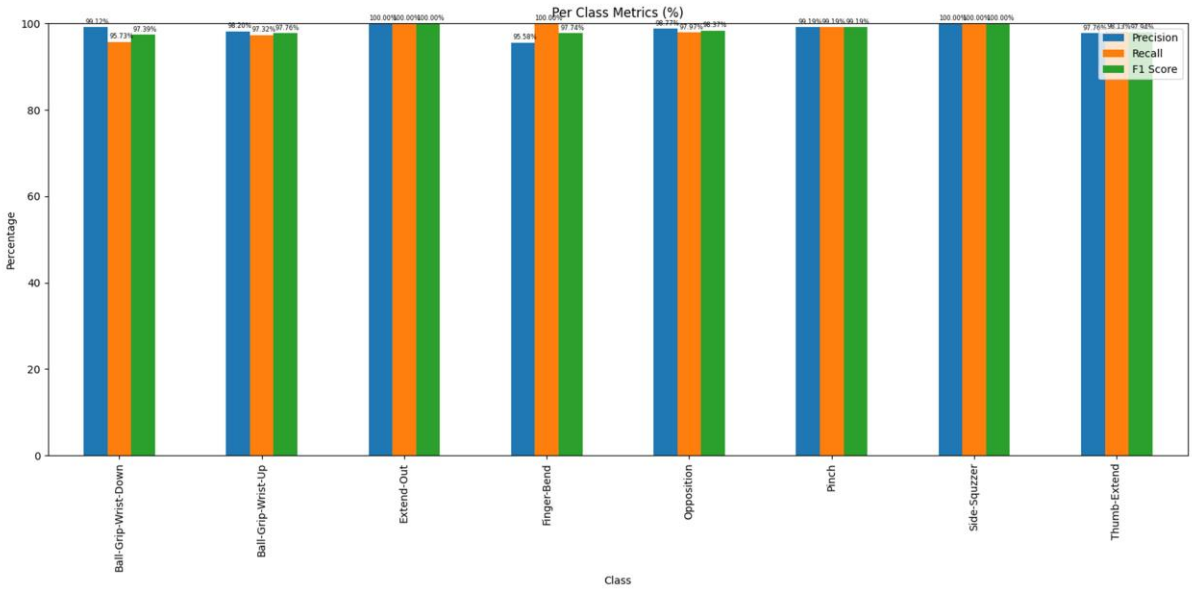
- The macro average of precision, recall, and F1 score is 99%, which indicates that each exercise class is equally considered in the evaluation.
- The weighted average also stands at 99%, confirming that the model performs consistently well even when considering the support (number of instances) for each exercise type.



4. Per-Class Performance: The classification report reveals detailed per-class performance. Each exercise type exhibits high precision, recall, and F1 score, confirming the model's effectiveness in distinguishing between different hand movements:

Table 4.4 Precision, Recall, and F1 Score per Exercise

Exercise	**Precision (%)**	**Recall (%)**	**F1 Score (%)**	**Support**
Ball-Grip-Wrist-Down	99	96	97	234
Ball-Grip-Wrist-Up	98	97	98	224
Extend-Out	100	100	100	267
Finger-Bend	96	100	98	238
Opposition	99	98	98	246
Pinch	99	99	99	247
Side-Squeezer	100	100	100	288
Thumb-Extend	98	98	98	267



4.2 Overall Model Performance

The Random Forest Classifier showed strong performance in classifying the various hand exercises, achieving an accuracy of 98.61%. This demonstrates that the model is highly effective in recognizing subtle differences in hand movements across all eight exercises. However, two important limitations were identified:

- Distance Sensitivity: The model's performance is influenced by the distance between the camera and the user's hand. Since the feedback mechanism relies on fixed reference points (e.g., distances between hand landmarks), variations in the camera's distance from the hand can affect the accuracy. If the user moves closer or farther from the camera, the system may misinterpret the hand positions, leading to incorrect feedback. This sensitivity could impact the reliability of real-time rehabilitation exercises.
- Smaller Hand Sizes: The model also struggles to maintain accuracy for users with smaller hands. Since the model relies on predefined reference points that assume average hand dimensions, users with smaller hands may experience less accurate feedback. This is because the distances between their hand landmarks may not align well with the expected reference points, resulting in potential errors during exercise classification.

Despite these limitations, the model's overall high accuracy indicates that it remains a reliable tool for most users. However, addressing these issues—such as by incorporating adaptive scaling for hand size and distance calibration—could further enhance the system's robustness and usability across a wider range of users.

4.3 Comparison with Different Machine Learning Models

In this study, we compared the performance of five widely used machine learning models: Logistic Regression, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Decision Tree, and Random Forest. Each of these models was evaluated on the task of recognizing hand exercises using the extracted hand landmarks. We measured their performance using accuracy, precision, recall, and F1 score, as shown in Table 4.5.

Among all the models, **Random Forest** consistently outperformed the others, achieving an accuracy of 98.61%, with precision, recall, and F1 scores also being the highest at 98.63%, 98.61%, and 98.61%, respectively. This superior performance can be attributed to several key advantages of the Random Forest algorithm, particularly in handling complex and non-linear datasets like the hand movements in our task.

Random Forest is an ensemble learning method, which means that instead of relying on a single decision tree, it builds multiple decision trees during training and combines their outputs to make a final prediction. This technique allows Random Forest to capture a broader range of patterns in the data. Specifically, it handles the intricate relationships between the various hand joints more effectively than simpler models like Logistic Regression, which assumes a linear relationship between features. This is important for hand exercise detection because hand movements involve highly non-linear patterns, especially when accounting for the subtle movements between fingers, palms, and wrists.

Another reason for Random Forest's superior performance is its ability to reduce overfitting. While models like **Decision Tree** often perform well on training data, they tend to overfit, leading to poorer generalization on new data. In contrast, Random Forest reduces this risk by averaging the predictions of multiple trees, each trained on different subsets of the data. This aggregation helps the model generalize better to unseen data, as evidenced by its higher accuracy and balanced precision and recall scores across all exercises. The high F1 score, which balances precision and recall, further confirms that Random Forest is not only accurate but also reliable in identifying both correct and incorrect hand movements.

In contrast, **Logistic Regression**, **SVM**, and **K-Nearest Neighbors** also performed well, but not as effectively as Random Forest. Logistic Regression and SVM assume simpler, often linear relationships between features, which limits their ability to fully capture the complex spatial patterns in hand gestures. Although K-Nearest Neighbors can model non-linear relationships, its performance depends heavily on the choice of neighbors, and it struggles with higher-dimensional data like the 21 hand landmark points we used. Meanwhile, Decision Tree, though flexible, is prone to overfitting, as seen in its lower accuracy of 93.98%.

In summary, **Random Forest** emerged as the best-performing model due to its ensemble approach, which allows it to handle non-linear relationships effectively, avoid overfitting, and generalize well to new data. This makes it an ideal choice for the complex task of hand exercise recognition, where subtle differences in hand movements need to be identified with high accuracy.

Table 4.5 Performance Comparison of Different Machine Learning Models

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
Logistic Regression	97.41	97.41	97.41	97.41
SVM	98.01	98.01	98.01	98.01
K-Nearest Neighbors	97.96	97.97	97.96	97.96
Decision Tree	93.98	94.00	93.98	93.98
Random Forest	**98.61**	**98.63**	**98.61**	**98.61**

4.4 Exercise-Specific Results

4.4.1 Result for Side Squeezer exercise

In figure 4.5 and 4.6, the program detects the user performing the Side Squeezer exercise and program checks how the user squeezes their fingers towards each other. It monitors the positions of the fingers and the force applied during the squeeze. If the user performs the exercise correctly, written feedback is displayed. Otherwise, the system instructs the user to adjust their squeezing technique. Images for both correct and incorrect postures have been attached, showing feedback on the live feed.

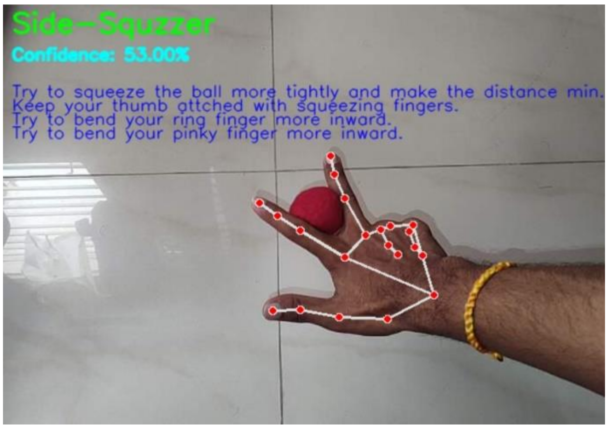
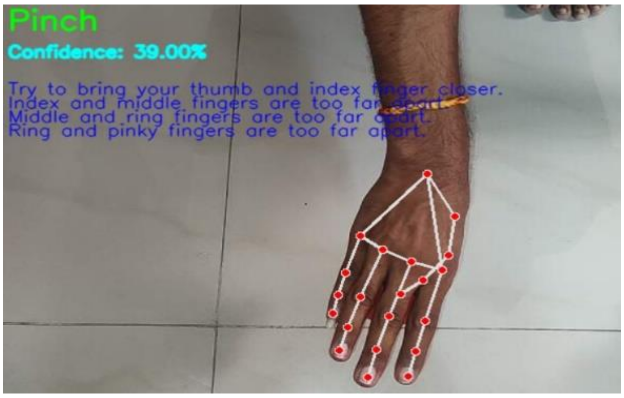


Figure 4.5 Performing Side Squzzer in wrong way   Figure 4.6 Performing Side Squzzer in correct way

4.4.2 Result for Pinch exercise

During the Pinch exercise, the system focuses on the position of the thumb and fingers. The model checks the distance between the thumb and other fingertips. If the user performs the pinch correctly, visual feedback is given on the screen. In case of improper form, the system provides written guidelines to adjust the hand posture. Images showing correct and incorrect pinch postures are attached, with feedback displayed on the live feed.



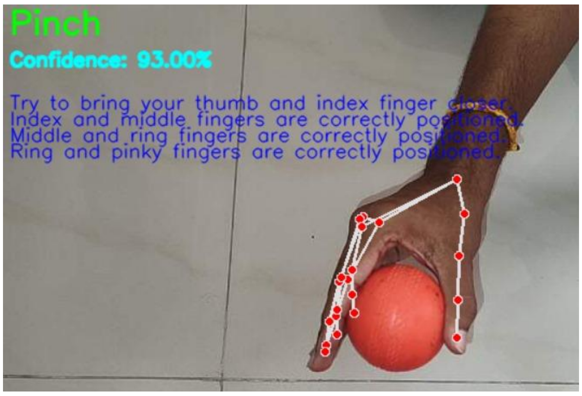


Figure 4.7 Performing Pinch in wrong way   Figure 4.8 Performing Pinch in correct way

4.4.3 Result for Extend Out exercise

The Extend Out exercise requires the user to fully extend all fingers outward. The system evaluates the distance between the fingers and tracks their position. When the user's hand posture matches the expected form, visual feedback is displayed. Any deviation, like insufficient finger extension, prompts written feedback for adjustment. Images displaying both correct and incorrect hand postures are attached, with feedback shown on the live feed.

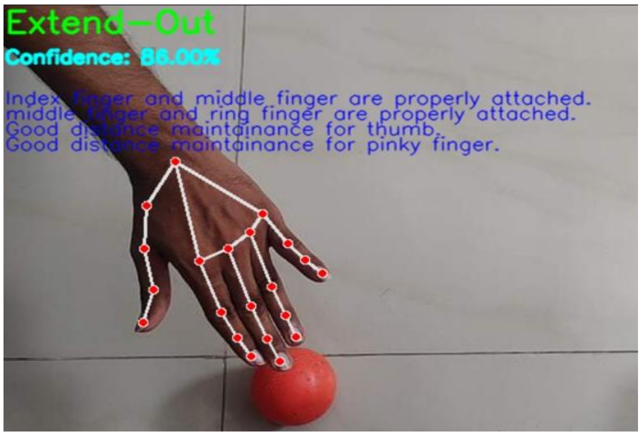
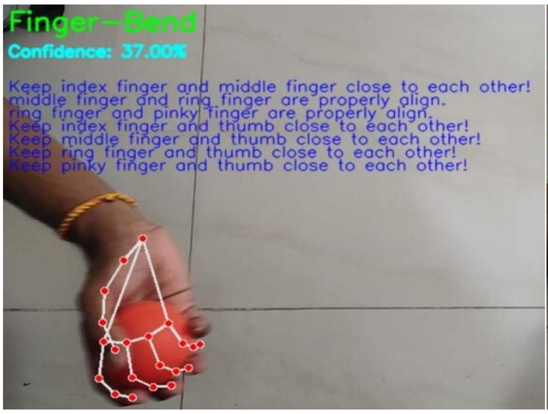


Figure 4.9 Performing Extend out in wrong way   Figure 4.10 Performing Extend Out in correct way

4.4.4 Result for Finger Bend exercise

For the Finger Bend exercise, the system tracks the bending angles of each finger. If the fingers are bent at the correct angles, the system shows confirmation on the screen. If the bending angle is either too shallow or too deep, the model provides feedback to adjust the finger positions. Attached images depict both correct and incorrect postures, with feedback visible on the live feed.



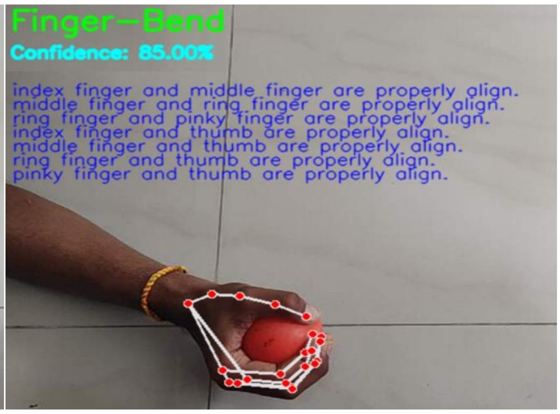


Figure 4.11 Performing Finger bend in wrong way   Figure 4.12 Performing Finger bend in wrong way

V. DISCUSSION

Using a Random Forest Classifier, the current study successfully developed a hand gesture recognition model that reached an accuracy of 98.61% on the test set. We demonstrate the benefits of our machine learning based approach to augmenting hand rehabilitation: patients with injuries, or in physical therapy, have access to accurate and actionable insights.

Given these features, what sets this work apart from previous research is the focus on an affordable, accessible, and user-friendly solution tailored to hand movements in comparison to wider systems of motor rehabilitation and expensive, complicated setups such as VR or robotic orthosis devices. The majority of previous attempts have focused on full or upper body rehabilitation and have ignored the subtle and intricate movement of the fingers, palms, and wrists. In the model we bridge the gap by utilizing MediaPipe hand tracking, which is embedded into a machine learning framework to make the hand detection and feedback more specific towards hand exercises.

Several key design choices and observations during research process made our model highly accurate. The first choice for classifier was Random Forest Classifier because it can process complex, nonlinear relationship between the hand landmarks recorded during the exercises. The model mitigates overfitting and improves generalization by using the ensemble learning nature of Random Forrest. Ensuring robustness and consistency of the model in real world, non-ideal conditions such as varying hand sizes or slight motion variation, required the selection of hyper parameters like number of estimators, and repeated testing across multiple random states to fine tune the model.

Our system, however, is not only able to guide the limb in response to hand movements, but to provide real time feedback, with operational characteristics superior to those of current approaches that often provide poor precision or responsiveness for delicate hand rehabilitation tasks. As opposed to conventional physiotherapy where you just get to go in for exercises which some therapists might be around for and some not, with this model you’re being continuously monitored and corrected by hand exercises. In addition, the system uses standard webcams and consumer laptops, furthering a low cost, inexpensive approach to home-based rehabilitation that does not require specialized hardware.

This system has some potential real world uses. The model can decrease the workload of physiotherapists for patients who do the exercises in hospitals or rehabilitation centers while guided by the machine and with very little supervision. For example, this automated system could additionally be used in home care settings where patients would be able to continue with their rehabilitation with precision feedback such that they are performing proper technique and don't injure themselves. The system improves rehabilitation outcomes in both cases by making patients execute prescribed exercises more accurately and consistently than is possible with manual methods.

And importantly, the model is ideal for environments where access to resources is limited, as is so often the case in rural or underfunded healthcare facilities, where there are few trained specialists and limited high end rehabilitation devices available. Integration of this system into existing healthcare workflow enables rehabilitation programs to be extended to a broader population and every part of the population to recover faster with better long-term outcome.

We show how machine learning can be effectively combined with non-invasive tool such as MediaPipe for hand specific rehabilitation which expands current knowledge. This goes beyond hand rehabilitation to suggest that other types of motor recovery or physical therapy may also benefit from similar machine learning frameworks, thus, adding to the body of personalized, technology driven healthcare solutions.

Future research should expand the model to include more hand exercises (or movement combinations) beyond static exercises. Assessment of the model performance in relation to the traditional therapy would be also required through clinical trials for further validation. Furthermore, applied to computation, advanced machine learning such as deep learning or reinforcement learning can further boost the precision and flexibility of the feedback system, to create another generation of smart rehabilitation technologies that bring even more utility to the patients and health care providers.

VI. CONCLUSION

In this work, we developed a hand gesture recognition system utilizing a Random Forest Classifier achieving an accuracy of 98.61 across eight separate hand exercises. The real feedback based on this simple, non-invasive setup (laptop, webcam) makes it possible to fill in the key gaps of existing hand rehabilitation solutions. Instead, our model is aligned with a focus on the precise detection of hand movements, presenting a more accessible and less expensive alternative to the conventional rehabilitation devices, which are moreover rather complex and expensive.

To accomplish this, we integrated MediaPipe for hand tracking, and with meticulous tuning of hyper parameters established a model that is not only accurate but feasible on general purpose computation devices without dedicated hardware. With its real time feedback capability, combined with its ease of use, this system makes it ideal for both clinical settings and home-based rehabilitation, allowing patients to immediately receive corrections without therapists around to supervise them. The ability for this advancement to reduce workload in hospitals and rehabilitation centers while increasing patient compliance and outcome is a sign of its advancement.

In contrast to previous work, we concentrate specifically on hand exercises with greater detail and propose a low-cost solution that seamlessly enhances daily routines. An important step forward for hand rehabilitation technology is the model's ability to track hand movements in a precise fashion and provide real time feedback, a clinical need (precision) and a practical need (affordability).

This model can be expanded to support other hand exercises and included in larger rehabilitation systems down the road. As with all new technology, the system will need to be proven to have clinical utility over traditional therapy methods in clinical trials. Likewise, more advanced machine learning techniques or deep learning approaches could together improve feedback precision and elasticity to prepare for broader deployment of personalized rehabilitation solutions.

REFERENCES

[1] I. Grubiˇsi´c, H. Skala Kavanagh, and S. Grazio, "Novel approaches in hand rehabilitation," Periodicum biologorum, vol. 117, no. 1, pp. 139–

[2] A. Rhoda, N. Cunningham, S. Azaria, and G. Urimubenshi, "Provision of inpatient rehabilitation and challenges experienced with participation post discharge: quantitative and qualitative inquiry of african stroke patients,"

[3] S. V. Adamovich, A. S. Merians, R. Boian, J. A. Lewis, M. Tremaine, G. S. Burdea, M. Recce, and H. Poizner, "A virtual reality—based exercise system for hand rehabilitation post-stroke," Presence: Teleoperators & Virtual Environments, vol. 14, no. 2, pp. 161–174, 2005.

[4] S. Ates, C. J. Haarman, and A. H. Stienen, "Script passive orthosis: design of interactive hand and wrist exoskeleton for rehabilitation at home after stroke," Autonomous Robots, vol. 41, pp. 711–723, 2017.

[5] R. A. Bos, C. J. Haarman, T. Stortelder, K. Nizamis, J. L. Herder, A. H. Stienen, and D. H. Plettenburg, "A structured overview of trends and technologies used in dynamic

hand orthoses," Journal of neuroengineering and rehabilitation, vol. 13, pp. 1–25, 2016.

[6] M. Muller, D. Tsui, R. Schnurr, L. Biddulph-Deisroth, J. Hard, and J. C. MacDermid, "Effectiveness of hand therapy interventions in primary management of carpal tunnel syndrome: a systematic review," Journal of Hand Therapy, vol. 17, no. 2, pp. 210–228, 2004.

[7] C. Arrowsmith, D. Burns, T. Mak, M. Hardisty, and C. Whyne, "Physiotherapy exercise classification with single-camera pose detection and machine learning," Sensors, vol. 23, no. 1, p. 363, 2022.

[8] P. Peer, A. Jakliˇc, and L. ˇSajn, "A computer vision based system for a rehabilitation of a human hand," Periodicum biologorum, vol. 115, no. 4, pp. 535–544, 2013.

[9] Y. Qui˜nonez, C. Lizarraga, and R. Aguayo, "Machine learning solutions with mediapipe," in 2022 11th International Conference On Software Process Improvement (CIMPS). IEEE, 2022, pp. 212–215.

[10] A. A. Zainuddin, N. H. Mohd Dhuzuki, A. A. Puzi, M. N. Johar, and M. Yazid, "Calibrating hand gesture recognition for stroke rehabilitation internet-of-things (riot) using mediapipe in smart healthcare systems." International Journal of Advanced Computer Science & Applications, vol. 15, no. 7, 2024.

[11] G. S´anchez-Brizuela, A. Cisnal, E. de la Fuente-L´opez, J.-C. Fraile, and J. P´erez Turiel, "Lightweight real-time hand segmentation leveraging mediapipe landmark detection," Virtual Reality, vol. 27, no. 4, pp. 3125–3132, 2023.

[12] E. Berglund, I. Jedel, and A. Berglund, "Using mediapipe machine learning to design casual exertion games to interrupt prolonged sedentary lifestyle," in International Conference on Human-Computer Interaction. Springer, 2023, pp. 237–251.

[13] V. Kriˇznar, M. Leskovˇsek, and B. Batagelj, "Use of computer vision based hand tracking in educational environments," in 2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO). IEEE, 2021, pp. 804–809.

[14] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. Yong, J. Lee et al., "Mediapipe: A framework for perceiving and processing reality," in Third workshop on computer vision for AR/VR at IEEE computer vision and pattern recognition (CVPR), vol. 2019, 2019.

[15] E. Regondi, "Development of an immersive augmented reality mobile experience: a study of ar mirroring effect," 2022.

[16] S. Metkar, J. Mahajan, J. Adsul, and B. Chavan, "Human body gesture-controlled gaming application," in 2022 Second International Conference on Next Generation Intelligent Systems (ICNGIS). IEEE, 2022, pp. 1–6.

[17] W. Wang, "Virtual conference with augmented technology," Ph.D. dissertation, WORCESTER POLYTECHNIC INSTITUTE.

[18] J.-W. Kim, J.-Y. Choi, E.-J. Ha, and J.-H. Choi, "Human pose estimation using mediapipe pose and optimization method based on a humanoid model," Applied sciences, vol. 13, no. 4, p. 2700, 2023.

[19] D. Bose, B. Arora, A. K. Srivastava, and P. Garg, "A computer vision based framework for posture analysis and performance prediction in athletes," in 2024 International Conference on Communication, Computer Sciences and Engineering (IC3SE). IEEE, 2024, pp. 942–947.

[20] G. Taware, R. Kharat, P. Dhende, P. Jondhalekar, and R. Agrawal, “Ai-based workout assistant and fitness guide,” in 2022 6th International Conference On Computing, Communication, Control And Automation (ICCUBEA. IEEE, 2022, pp. 1–4.
[21] F. Roggio, S. Di Grande, S. Cavalieri, D. Falla, and G. Musumeci, “Biomechanical posture analysis in healthy adults with machine learning: Applicability and reliability,” Sensors, vol. 24, no. 9, p. 2929, 2024.
[22] D. Ciraolo, M. Fazio, R. S. Calabr`o, M. Villari, and A. Celesti, “Facial expression recognition based on emotional artificial intelligence for tele-rehabilitation,” Biomedical Signal Processing and Control, vol. 92, p. 106096, 2024.
[23] T. Banerjee, “Computer vision-based hand tracking and 3d reconstruction as a human computer input modality with clinical application,” Master’s thesis, The University of Western Ontario (Canada), 2023.
[24] J.-S. Han, C.-I. Lee, Y.-H. Youn, and S.-J. Kim, “A study on real-time hand gesture recognition technology by machine learning-based mediapipe,” Journal of System and Management Sciences, vol. 12, no. 2, pp. 462–476, 2022.
[25] S. Prema, G. Deena, D. Hemalatha, K. Aruna, and S. Hashini, “Gaming using different hand gestures using artificial neural network,” EAI Endorsed Transactions on Internet of Things, vol. 10, 2024.
[26] D. Reimer, I. Podkosova, D. Scherzer, and H. Kaufmann, “Evaluation and improvement of hmd-based and rgb-based hand tracking solutions in vr,” Frontiers in Virtual Reality, vol. 4, p. 1169313, 2023.
[27] C. Budzinski, H.-L. Wu, E. Sarraf, S. Miller, and J. Moore, “Accuracy of mediapipe visual hand tracking for use in medical training procedures,” in Frontiers in Biomedical Devices, vol. 87752. American Society of Mechanical Engineers, 2024, p. V001T08A004.
[28] O. N. Zestas, D. N. Soumis, K. D. Kyriakou, K. Seklou, and N. D. Tselikas, “A computer-vision based hand rehabilitation assessment suite,” AEU-International Journal of Electronics and Communications, vol. 169, p. 154762, 2023.
[29] Y. Kwon and D. Kim, “Real-time workout posture correction using opencv and mediapipe,” , vol. 20, no. 1, pp. 199–208, 2022.
[30] Y. Zhang, “Applications of google mediapipe pose estimation using a single camera,” 2022.
[31] P.-C. Hsiao, S.-Y. Yang, C.-H. Ho, W. Chou, and S.-R. Lu, “The benefit of early rehabilitation following tendon repair of the hand: A population-based claims database analysis,” Journal of hand therapy, vol. 28, no. 1, pp. 20–26, 2015.
[32] S. D. Horn, G. DeJong, and D. Deutscher, “Practice-based evidence research in rehabilitation: an alternative to randomized controlled trials and traditional observational studies,” Archives of Physical Medicine and Rehabilitation, vol. 93, no. 8, pp. S127–S137, 2012.
[33] N. R. Council, D. of Behavioral, S. Sciences, B. on Human-Systems Integration, and C. on the Role of Human Factors in Home Health Care, Health care comes home: the human factors. National Academies Press, 2011.
[34] M. A. Delph, S. A. Fischer, P. W. Gauthier, C. H. M. Luna, E. A. Clancy, and G. S. Fischer, “A soft robotic exomusculature glove with integrated semg sensing for hand rehabilitation,” in 2013 IEEE 13th International Conference on Rehabilitation Robotics (ICORR). IEEE, 2013, pp. 1–7.

[35] B. Radder, A. Kottink, N. Van Der Vaart, D. Oosting, J. Buurke, S. M. Nijenhuis, G. B. Prange, and J. S. Rietman, “User-centred input for a wearable soft-robotic glove supporting hand function in daily life,” in 2015 IEEE International Conference on Rehabilitation Robotics (ICORR). IEEE, 2015, pp. 502–507.
[36] V. Vijayan, J. P. Connolly, J. Condell, N. McKelvey, and P. Gardiner, “Review of wearable devices and data collection considerations for connected health,” Sensors, vol. 21, no. 16, p. 5589, 2021.
[37] M. F. Pereira, C. Prahm, J. Kolbenschlag, E. Oliveira, and N. F. Rodrigues, “Application of ar and vr in hand rehabilitation: A systematic review,” Journal of Biomedical Informatics, vol. 111, p. 103584, 2020.
[38] C. Arrowsmith, D. Burns, T. Mak, M. Hardisty, and C. Whyne, “Physiotherapy exercise classification with single-camera pose detection and machine learning,” Sensors, vol. 23, no. 1, p. 363, 2022.
[39] S. Rana, M. Singh, R. A. Choudhury, and N. S. Chawla, “Hand tracking for rehabilitation using machine vision,” Kilby, vol. 100, p. 7th, 2023.
[40] B. Thaman, T. Cao, and N. Caporusso, “Face mask detection using mediapipe facemesh,” in 2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO). IEEE, 2022, pp. 378–382.