Master of Technology in Artificial Intelligence Systems

# PROJECT REPORT

## Motion Analysis


| Group 10 | | |
|---|---|---|
| **Name** | **Student ID** | **Email** |
| Jin Keyi | e1133134@u.nus.edu | A0276819L |
| Ko Hung-Chi | e1539175@u.nus.edu | A0327344E |
| Sun Yuchen | e1538079@u.nus.edu | A0326248B |
| Zhang Yuxuan | e1216649@u.nus.edu | A0285664N |
| Zhao Jiahui | e1554179@u.nus.edu | A0329852U |

# Contents

**1 Executive Summary** **3**

**2 Project Descriptions** **3**

**3 Methodology** **3**

3.1 Dataset Preparation . . . . . 4

3.2 Data Preprocessing . . . . . 4

3.3 System models . . . . . 5

3.4 System performance . . . . . 6

3.5 Motion correction . . . . . 7

**4 Conclusion** **8**

# 1 Executive Summary

Physical exercise is essential for maintaining health, yet improper form during workouts often leads to injuries and reduced training effectiveness. To address this challenge, our team developed **Motion Analysis System** that provides objective, data-driven feedback on exercise performance.

By combining artificial intelligence with 3D motion data from the **Fit3D dataset**, our solution automatically identifies posture deviations, enabling correction and improvement suggestions.

The methodology involves three main components: (1) **motion feature learning** using an autoencoder to extract compact representations of exercise sequences; (2) **exercise classification** through a GRU-based model trained on labeled motion data; and (3) **system integration** that evaluates both existing and new motion data to provide posture assessment.

The results demonstrate that the proposed approach can accurately classify multiple types of exercise and highlight deviation regions, supporting safer and more efficient training. The system shows potential for applications in smart gyms, home fitness monitoring, and rehabilitation contexts.

# 2 Project Descriptions

**Why we designed this system:** Proper exercise execution is essential for maintaining health and achieving optimal performance, but incorrect posture in gyms often leads to injuries and hinders progress. This project evaluates exercise form and provides feedback for improvement. It aims to automate exercise recognition and posture deviation detection, enabling data-supported, repeatable training guidance.

**How the system solves the problem:** The system uses the **Fit3D dataset**, which offers 3D human pose and shape data across multiple exercise types. Our approach follows a three-stage pipeline: (1) motion feature learning through an autoencoder that extracts compact representations of body movements; (2) exercise classification using a GRU-based sequence model trained on the extracted features; and (3) system integration that evaluates both existing and new motion data to provide real-time feedback on posture accuracy. This workflow allows the system to identify movement types, measure form deviations, and deliver actionable feedback to support safer and more effective workouts.

# 3 Methodology

Our processing pipeline proceeds in: data extraction, model training and evaluation, and new-data inference.

First, we perform data preparation by selecting four target exercise classes from the Fit3D dataset — *dumbbell hammer curls*, *dumbbell reverse lunge*, *neutral overhead shoulder press*, and *push-up*.

For each selected sequence in the dataset we extract the per-frame human body parameters (pose, shape, skeleton) and store the results. This extraction aligns the raw multi-view and model-based annotations of Fit3D with our downstream modules.

Next, we proceed to unsupervised feature learning and supervised classification. Using the extracted features from the training set, an autoencoder is trained in an unsupervised manner to learn a compact latent representation of the motion features. Once the latent space is established, a GRU-based model is trained to classify each repetition into one of the four exercise types.

For evaluation, a test set is processed with the same feature-extraction pipeline, and the trained models are applied for classification and reconstruction of the unseen test sequences. We also use new data: three people each performing the same four exercises in videos. These recordings are processed through the extraction pipeline and then fed into the trained classifier to test model generalization and to validate system robustness outside the original dataset.

## 3.1 Dataset Preparation

Our system uses **the Fit3D dataset** as the core dataset. The key characteristics of the dataset are as follows:

- The Fit3D Dataset comprises 611 multi-view sequences, each containing a minimum of 5 annotated exercise repetitions per sequence and highly accurate ground-truth 3D skeletons, along with human pose and shape parameters based on models such as GHUM and SMPLX.
- The collection supports 37 types of exercises, covering all major muscle groups, and is performed by both instructors and trainees, enabling robust training and evaluation of 3D human motion and shape reconstruction under exercise conditions.
- The dataset is designed to support research on 3D human sensing for fitness training, aiming to reconstruct 3D human pose, shape, and motion, reliably segment exercise repetitions, and provide quantitative real-time feedback on deviations from instructor standards to reduce injury risk and support continuous improvement.

## 3.2 Data Preprocessing

Before training, all Fit3D recordings were preprocessed to convert raw 3D tracking data into structured, labeled sequences suitable for model learning. Each recording contains synchronized multi-view videos and frame-wise SMPL-X parameters, describing the subject's 3D body pose, orientation, and shape throughout the motion.

First, the tracking data and videos were temporally aligned, ensuring one-to-one correspondence between 3D poses and video frames. The motion sequences were then analyzed to detect exercise repetitions using kinematic features such as joint motion energy and novelty variation. Peaks and valleys in these features were used to automatically determine repetition boundaries, isolating individual movement cycles (e.g., one full push-up or squat).

Each repetition segment was standardized to a uniform temporal length of 150 frames to ensure consistency across samples. Shorter sequences were padded, and longer ones truncated, preserving the natural motion dynamics. The resulting segments were assigned exercise labels according to their original categories, including major movements such as *push-up*, *squat*, *curl*, and a general *other* class for residual actions.

Finally, to verify the quality of segmentation, 3D joint trajectories were reprojected onto the original video frames for visual inspection. This confirmed the correctness of temporal alignment between the reconstructed poses and the raw visual motions. Through this pipeline, complex, variable-length 3D tracking sequences were transformed into uniform, labeled motion samples, providing a clean and reliable foundation for model training and evaluation.

### 3.3 System models

Our system employs a two-stage model architecture that combines unsupervised feature learning with supervised classification. Each model component plays a complementary role in recognizing and understanding human motion patterns.

**1. Autoencoder for Motion Feature Learning:** The first stage of the pipeline employs an **autoencoder** trained in an unsupervised fashion to learn a compact and discriminative latent representation of motion features derived from the SMPL-X parameters.

- **Input:** Flattened per-frame SMPL-X pose features (approximately 201 dimensions), normalized across the entire dataset.
- **Encoder:** A stack of fully-connected layers with ReLU activations progressively reduces the input dimensionality to a lower-dimensional latent vector.
- **Decoder:** A mirrored set of layers reconstructs the original feature vector from the latent space.
- **Loss Function:** Mean Squared Error (MSE) between the reconstructed and input feature vectors is minimized:
$$\mathcal{L}_{AE} = \frac{1}{N} \sum_{i=1}^{N} \|x_i - \hat{x}_i\|_2^2$$

This autoencoder serves two key purposes: (i) it denoises and compresses noisy high-dimensional pose features into a stable latent representation; and (ii) it supports downstream classification by providing embeddings that capture the essential structure of human motion.

**2. GRU-based Motion Classification Model:** The second stage applies a **Gated Recurrent Unit (GRU)** network to capture the temporal dependencies inherent in sequential pose data.

- **Input:** Sequences of latent embeddings from the autoencoder, representing the temporal evolution of motion.
- **Architecture:** One or more stacked GRU layers followed by dropout and a fully-connected softmax layer to predict the exercise class.
- **Loss Function:** Cross-entropy loss is used to train the classifier:

$$\mathcal{L}_{CE} = -\frac{1}{N}\sum_{i=1}^{N} y_i \log(\hat{y}_i)$$

- **Optimization:** Adam optimizer with adaptive learning rate scheduling ensures stable convergence.

GRU units effectively capture temporal continuity and repetition patterns in exercises such as push-ups or lunges, outperforming static classifiers on time-dependent features.

**3. Model Integration and Workflow:** The two models operate sequentially during both training and inference:

1. Extract pose features from Fit3D sequences.
2. Train the autoencoder on all classes to obtain compact motion embeddings.
3. Feed these embeddings into the GRU-based classifier for supervised training.
4. During inference, unseen motion sequences are encoded and classified through the same pipeline.

This modular design supports flexibility — the autoencoder may later be replaced by other feature encoders (e.g., CNNs or Transformers) without altering the classification logic. Moreover, reconstruction loss from the autoencoder can serve as an anomaly indicator, identifying out-of-distribution or incorrect motion patterns.

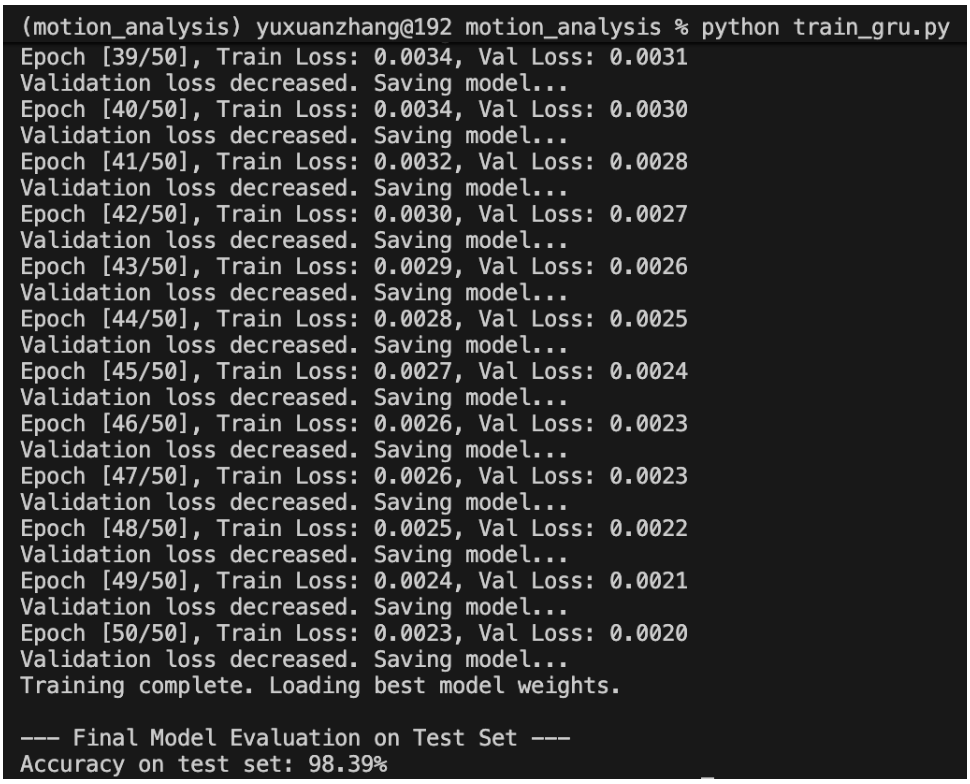
## 3.4 System performance

The performance of the developed motion classification system was evaluated using a GRU-based sequential model trained on the processed motion dataset. During training, both the training and validation losses showed a clear downward trend, indicating stable convergence and strong learning behavior.

Before model training, visualization experiments were carried out to verify the correctness of the motion tracking and segmentation results. Reconstructed 3D joint trajectories were projected onto the corresponding video frames, ensuring that each preprocessed repetition segment accurately represented a complete and valid exercise cycle. This step confirmed the quality and reliability of the dataset used for model training.

The final evaluation demonstrated high overall accuracy and consistent classification performance across various exercise categories. The GRU network effectively captured temporal motion dependencies from 3D pose sequences, enabling accurate recognition of exercise patterns and repetition structures.

Overall, the system achieved robust and accurate motion analysis results, reflecting the effectiveness of the data preprocessing, visualization validation, and sequential modeling pipeline.



```
(motion_analysis) yuxuanzhang@192 motion_analysis % python train_gru.py
Epoch [39/50], Train Loss: 0.0034, Val Loss: 0.0031
Validation loss decreased. Saving model...
Epoch [40/50], Train Loss: 0.0034, Val Loss: 0.0030
Validation loss decreased. Saving model...
Epoch [41/50], Train Loss: 0.0032, Val Loss: 0.0028
Validation loss decreased. Saving model...
Epoch [42/50], Train Loss: 0.0030, Val Loss: 0.0027
Validation loss decreased. Saving model...
Epoch [43/50], Train Loss: 0.0029, Val Loss: 0.0026
Validation loss decreased. Saving model...
Epoch [44/50], Train Loss: 0.0028, Val Loss: 0.0025
Validation loss decreased. Saving model...
Epoch [45/50], Train Loss: 0.0027, Val Loss: 0.0024
Validation loss decreased. Saving model...
Epoch [46/50], Train Loss: 0.0026, Val Loss: 0.0023
Validation loss decreased. Saving model...
Epoch [47/50], Train Loss: 0.0026, Val Loss: 0.0023
Validation loss decreased. Saving model...
Epoch [48/50], Train Loss: 0.0025, Val Loss: 0.0022
Validation loss decreased. Saving model...
Epoch [49/50], Train Loss: 0.0024, Val Loss: 0.0021
Validation loss decreased. Saving model...
Epoch [50/50], Train Loss: 0.0023, Val Loss: 0.0020
Validation loss decreased. Saving model...
Training complete. Loading best model weights.

--- Final Model Evaluation on Test Set ---
Accuracy on test set: 98.39%
```

Figure 1: performance

## 3.5 Motion correction

To make the system usable in a real application scenario, a motion correction pipeline was designed to process user-uploaded videos and automatically assess the quality of the extracted 3D poses. Instead of relying only on pre-collected, well-aligned motion data, the system accepts raw SMPL-X parameter sequences converted from a user's video, including translation, global orientation, and body pose matrices. These parameters are first normalized and converted to a unified format (150×21×3) so that videos of different lengths and capture conditions can be evaluated in a consistent way.

After preprocessing, the sequence is fed into the LSTM autoencoder that was previously trained on standard motion patterns. The autoencoder reconstructs the whole 150-frame motion, and the reconstructed joints are compared with the original ones to obtain joint-wise angular errors. Unlike a single global threshold, the system adopts per-joint adaptive thresholds (set to 1.5× the mean error of that joint across the sequence), so that naturally more flexible or noisy joints do not falsely dominate the error decisions. Based on this rule, the system produces a 150×21 boolean matrix, where each entry directly indicates whether a specific joint at a specific frame should be considered reliable (True) or abnormal (False).

For interpretability, the corrected result can be visualized together with the original video: valid joints are rendered in green and invalid ones in red on top of the video frames. This makes it easy for an end user to see where the pose tracking failed (e.g., wrist jitter, ankle drift, shoulder flip) and when it happened in the motion.

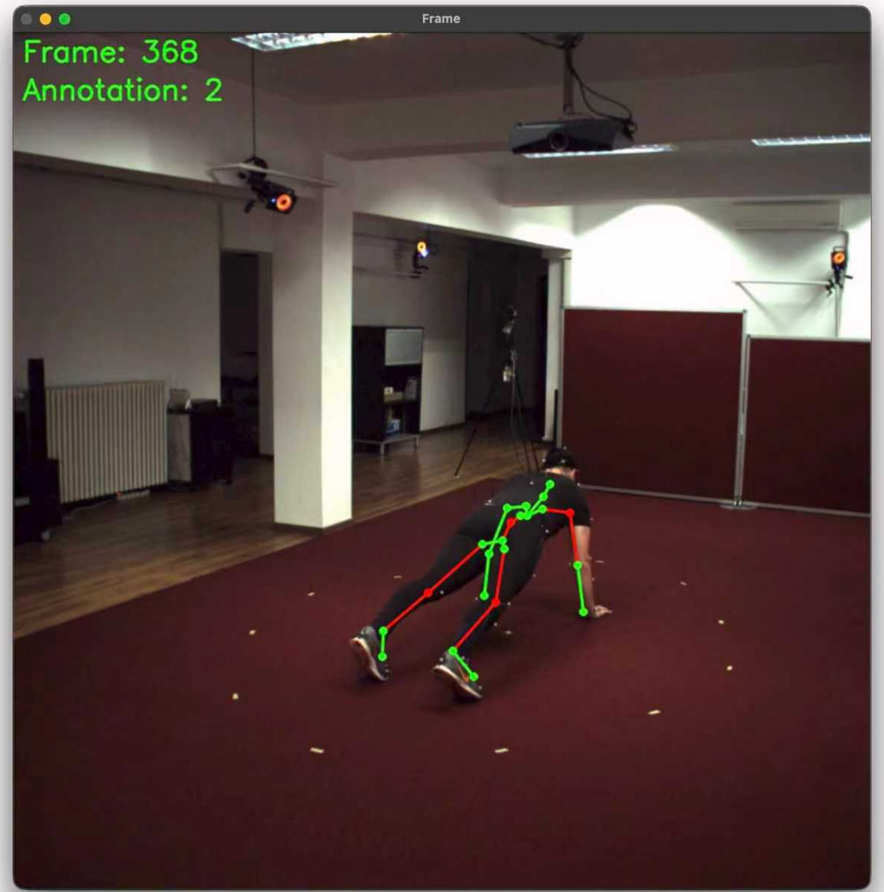



Figure 2: Correction Visualization

# 4 Conclusion

In conclusion, motion analysis demonstrates the application of pattern recognition and modeling to human motion understanding in gym environments. Using the Fit3D dataset, which provides accurate 3D skeletal and shape data across multiple types of exercise, our system integrates motion recognition and correction within a unified framework. Through this project, we demonstrate the potential of combining artificial intelligence sensing to enhance exercise performance and mitigate injury risks.

Overall, this work highlights how pattern recognition can support the development of smart fitness solutions, remote physiotherapy, and wearable devices that provide feedback. Future extensions include incorporating multi-person tracking, equipment-based exercises, and real-time deployment through lightweight model optimization to achieve a comprehensive, real-world AI fitness assistant.