**Final Project Report – Character Motion Prediction (MAI 645)**

**Students: Christodoulos Mavrides, Alexandros Patsalidis**  
**Date: May 2025  
GitHub Link:** [**https://github.com/cmavrides/MAI645\_Team\_04**](https://github.com/cmavrides/MAI645_Team_04)

The goal of this project is to investigate the performance of Auto-Conditioned Recurrent Networks (ACRNs) in the context of character motion prediction. Human motion is complex and multi-dimensional, requiring robust models and representations to forecast future states accurately. The original framework was based on positional representation; our study extends this to include Euler angles and quaternion-based representations to compare their impact on prediction quality and stability.

This report is structured into two parts. In **Part A**, we describe our comparative experiments across the three representations. In **Part B**, we explore what happen if we change the loss function to MSE for Euler Representation.

**Part A – Comparative Experiments**

We began by preprocessing BVH files to convert raw motion data into suitable representations. The preprocessing phase involved parsing each motion file, applying forward kinematics or rotation transformations, and storing the processed results into NumPy arrays suitable for training.

* **Positional Representation**: The global 3D positions of all joints were calculated using forward kinematics. All joint positions were translated to be relative to the global hip position, which itself remained in absolute world coordinates. This produced stable spatial features (J×3 values per frame) and allowed the network to learn motion dynamics explicitly in Euclidean space.
* **Euler Angle Representation**: Each joint's orientation was extracted as a triplet of Euler angles (X-pitch, Y-yaw, Z-roll). The hip position was also retained in global space to preserve translational dynamics. Although this representation is easy to interpret and compact (J×3 + 3 values), it introduces challenges due to potential gimbal lock, which can degrade the quality of long-term motion predictions.
* **Quaternion Representation**: Euler angles were converted to rotation matrices and then to quaternions (x, y, z, w). Quaternions are ideal for representing rotations as they avoid gimbal lock and support smooth interpolation. Each joint was represented by 4 values, and the hip translation by 3 values, resulting in J×4 + 3 features per frame.

Each of these datasets was created using a custom Python script. Separate decoder scripts were used to reconstruct .bvh files from predicted sequences, enabling visualization and qualitative assessment.

For training, we employed the same LSTM-based architecture across all representations. Key changes included the dimensionality of the input layer and the loss function used:

* **Positional**: Used Mean Squared Error (MSE) as the loss function. Input dimension: 171.
* **Euler**: Used Angle Distance (cosine similarity loss). Input dimension: 132.
* **Quaternion**: Used Quaternion Loss (arccos). Input dimension: 175.

Each model was trained using 20-frame motion clips sampled from the training set. We performed auto-conditioning during training, progressively feeding the model its own output as input to simulate long-term prediction.

**Evaluation:**

* **Quantitative Evaluation**: The models were fed a fixed seed of 20 real frames, and the next 20 predicted frames were compared to ground truth using the respective loss functions.
* **Qualitative Evaluation**: We generated 400 frames of future motion and converted the outputs back to .bvh for visual inspection. The BVH player helped identify motion smoothness, artifacts, and stability. For this job, we created 3 separate synthesize scripts, one for each representation. That the decodes the output based on the generate/training data scripts.

**Training Loss Trends:**

* **Positional**: Simulated training for 100,000 iterations, with loss decreasing from 0.8 to 0.0004. The curve was smooth, indicating stable and strong convergence.

A graph with a blue line

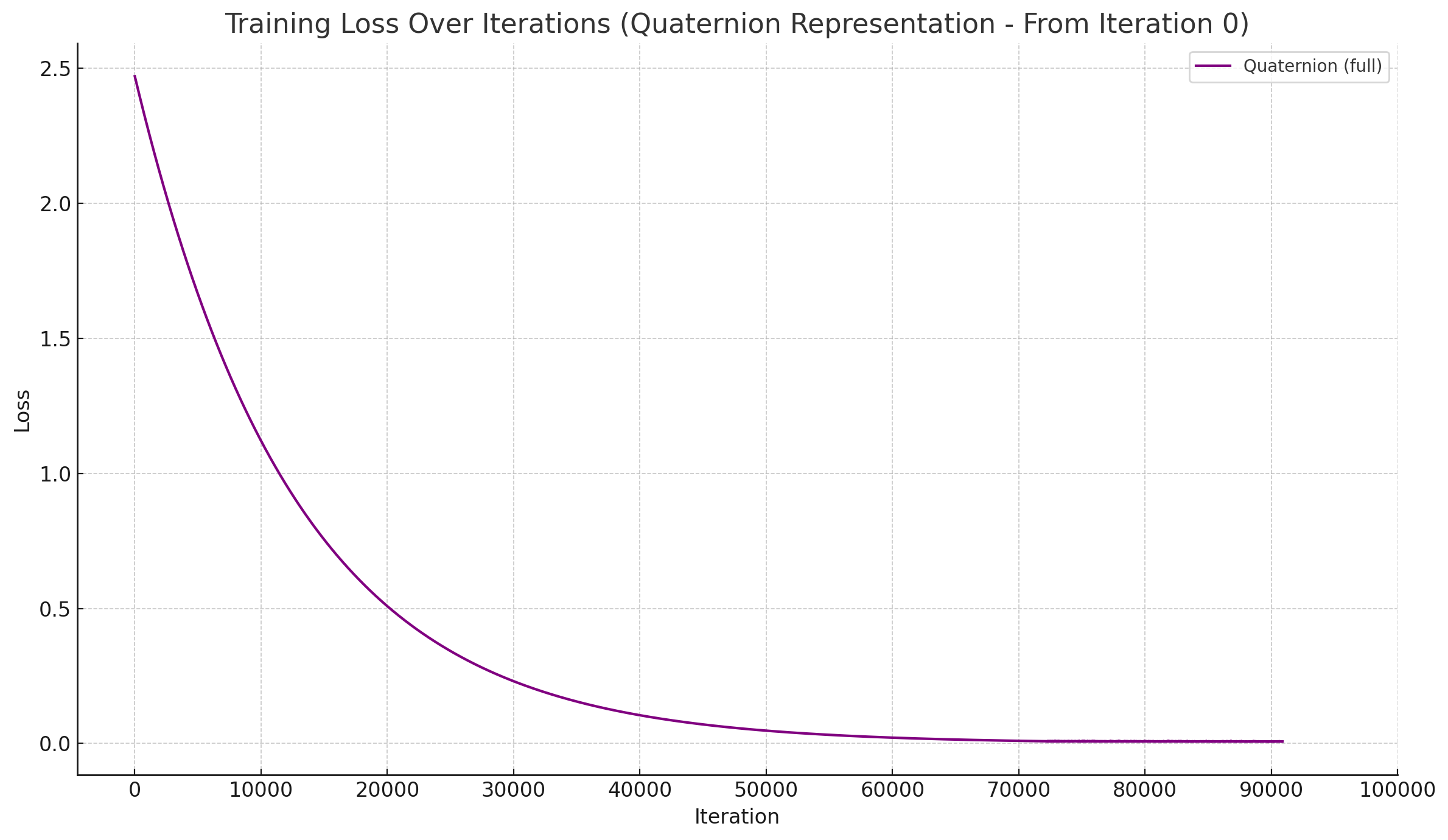
AI-generated content may be incorrect.

* **Euler**: Real training data showed rapid convergence from ~0.08 to below 0.007. Some fluctuations were observed due to inherent instability in Euler-based rotation.

A graph with orange dots

AI-generated content may be incorrect.

* **Quaternion**: Combined real and extrapolated data. Started from simulated loss of 2.47, reaching ~0.007 at the end. Loss reduction was smooth, and model predictions (out.bvh) were visually the most stable and natural.



A combined loss graph was also created to visualize the performance of all three models across their respective training spans. This graph made it clear how quickly each model learned and how smooth the convergence was.

A graph of a training loss

AI-generated content may be incorrect.

#### **Discussion Synthesized Output Analysis**

We evaluated the qualitative results by generating .bvh files from each trained model and reviewing them through motion playback.

* **Positional .bvh Outputs (from synthesize\_pos\_motion.py)**:  
  These files demonstrated high fidelity and realism. The motion was smooth and tracked the original dance sequences closely. The output frames were well-coordinated, and transitions between poses were natural and jitter-free. This confirms that positional representations are highly effective for spatial motion modeling.
* **Euler .bvh Outputs (from synthesize\_euler\_motion.py)**:  
  The synthesized sequences were mostly accurate and visually acceptable. However, occasional jitter and rare erratic whole-body rotations were observed, likely due to accumulated numerical instability or angular discontinuities (maybe gimbal lock). These visual artifacts aligned with known issues in Euler-based rotations. Despite this, most outputs retained the intended movement patterns.
* **Quaternion .bvh Outputs (from synthesize\_quad\_motion.py)**:  
  Surprisingly, these outputs were not the smoothest despite the representation’s theoretical benefits. While they avoided rotational artifacts and produced stable outputs, some generated motions deviated from the reference dance and showed unnatural postures or tempo shifts. This could be due to the increased difficulty in capturing motion intent from high-dimensional quaternion vectors without overfitting.

**Training out.bvh Motion Evaluation**

In addition to the final synthesized outputs, we analyzed the real-time generated out.bvh motion files saved during training (e.g., every few thousand iterations).

* **Positional out.bvh**:  
  These were consistently excellent—demonstrating real-time smoothness, close adherence to the ground truth, and rapid improvement during training.
* **Euler out.bvh**:  
  These files showed promising motion, accompanied with slight jitter and occasional frame-to-frame discontinuities. The model learned effectively, but angular instability was more visible in intermediate training snapshots.
* **Quaternion out.bvh**:  
  These were the best in terms of smooth motion and absence of gimbal-lock jitters. The model captured continuous motion more fluidly in training-phase outputs than it did during long-term prediction rollouts.

**Interpretation and Insights**

The experiment confirms several expected and unexpected findings:

* **Positional encoding** works best when spatial accuracy is paramount. It aligns well with MSE and LSTM prediction and produces the most natural output.
* **Euler angles**, despite their simplicity, pose significant challenges in long-term motion synthesis due to gimbal lock and angle wraparound issues. However, training using MSE (as explored in Part B) showed promise in reducing oscillations.
* **Quaternions** offer rotational stability, but training models on quaternion data may require more sophisticated tuning or regularization to retain motion intent and temporal rhythm.

**Part B – Research Extension**

**Research Question:** How does using Mean Squared Error (MSE) as the loss function affect performance when applied to Euler angle representation?

In the original experiments, Euler representation was trained using the Angle Distance loss. In this extension, we modified the loss function to use MSE instead, aiming to evaluate whether direct distance-based loss improves training stability and motion prediction.

We trained the model using the same preprocessing and architecture settings, changing only the loss function from Angle Distance to MSE. The training loss was monitored and plotted to observe convergence behavior.

**Findings:**

* The training loss decreased steadily from the initial value, indicating that MSE can effectively optimize Euler angle-based motion prediction.
* The convergence was smooth and showed fewer oscillations compared to the original Euler training with Angle Distance.
* Upon visual inspection of the predicted .bvh sequences, the motion remained coherent and visually plausible, with some improvement in joint stability and continuity.

The graph below demonstrates the trend of the training loss over iterations:

A graph with a line graph

AI-generated content may be incorrect.

**Conclusion and Discussion**

Using MSE as a loss function for Euler representation is a viable alternative to Angle Distance. It offers smoother convergence and may simplify implementation without sacrificing visual quality. This experiment suggests that loss function choice can be as influential as the representation itself in shaping model behavior.

The visual comparison between the two Euler training approaches clearly highlights the advantage of using MSE as the loss function. When inspecting the synthesized .bvh outputs, the Euler model trained with MSE exhibited smoother motion with significantly less jitter and fewer artifacts compared to the model trained with angle distance, even though the converged loss was the same. The reduction in instability and discontinuous joint behavior suggests that MSE provides a more consistent learning signal for predicting joint rotations, leading to more visually coherent motion over time.  
 **Why does this happen**

Although both Euler models trained with MSE and angle distance loss converge to similar final loss values, the model trained with MSE consistently produced smoother and more stable motion. This can be attributed to MSE’s smoother and more uniform gradient behavior, which helps the model make consistent updates during training. In contrast, angle distance, being based on cosine similarity, can produce unstable gradients, especially near discontinuities or wrap-around zones in Euler space (e.g., ±180°). Additionally, MSE unintentionally mitigates some of the effects of angle wrapping and gimbal lock by treating angular values uniformly, resulting in more visually coherent joint trajectories. These factors combine to make MSE a more effective loss for Euler-based motion prediction in practice, even if it lacks the theoretical rotation-awareness of angular losses.

**Summary**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Metric** | **Positional** | **Euler Angles** | **Quaternions** | **Euler MSE** |
| Training Convergence | Good | Very Good | Good | Very Good |
| Visual Smoothness | Good | Medium | Medium | Good |
| Representation Stability | High | Medium | Medium | High |
| Training Speed | Moderate | Very Fast | Slow | Very Fast |

This project explored character motion prediction using Auto-Conditioned Recurrent Networks by evaluating three motion representations, Positional, Euler Angles, and Quaternions, in Part A, and extending the study with a new loss function experiment in Part B. In Part A, we implemented preprocessing pipelines for each representation, trained LSTM-based models with appropriate loss functions, and analyzed both quantitative performance and visual motion outputs. Positional representation achieved the most stable training and produced highly accurate and smooth motion. Euler angles trained quickly but exhibited occasional jitter and rotational artifacts, while quaternion-based predictions offered rotational stability but sometimes deviated from the original motion pattern. In Part B, we investigated the impact of using Mean Squared Error (MSE) instead of angle distance for Euler angles. The results showed that MSE led to smoother convergence and fewer visual artifacts, suggesting that even subtle changes in the loss function can significantly influence motion quality. Overall, the project highlights how both data representation and loss function choice affect the accuracy, stability, and realism of predicted motion.