CSE 586 Final Project Narration

Body Pose Forecasting

Amrutha Somasundar, Himashveta Kumar

26th April 2025

PennState

Recap of Project Progress

Data preparation

- **Data Processing:**
  - For each .npz file
    - Load the file and skip the global rotation and the hand/finger joints, keeping only the main body joints.
    - Encode the 63D body poses into a 32D latent vector using VPoser
    - Downsample the data. We select every 4th frame in the sequence, thus reducing our frame rate to 30 FPS. 120 FPS might be too much data to process and most human motions don't require millisecond level tracking.
- **Training:**
  - We were planning to use subjects 1, 2, 6, 12 and 13 for training and subject 5 for testing.
  - Update: we used subjects 1, 2, 6 and 12 for training, 13 for validation, and 5 for testing.

PennState

Prediction horizon and Loss function

Timing	Frames at 30 FPS	Total Duration
Input	7 frames	0.25 seconds
Output	30 frames	1.0 seconds

- Human motion has an average reaction time of ~200 to 250ms. Short-term pose changes like walking stride transitions occur in ~300ms windows. We want to use the motion intent for prediction.
- For the output, 1 second gives real-world applicability. Beyond 1 second, human behavior becomes stochastic and unpredictable.
- To train our model, we will use Mean Squared Error (MSE).
- To evaluate the 3D joint accuracy, we will use Mean per Joint Position Error (MPJPE).

PennState

Baseline Method 1 for our prediction

- **Constant Position Model:**
  - The constant position model assumes that the body pose at time *t+1* will be the same as the pose at time *t*.
  - When evaluated the accuracy of the model, we received a MPJPE error of approximately 59 mm for a 1 second prediction

PennState

Next steps

Baseline Method 2 for our prediction

- **Constant Velocity Model:**
  - The Constant Velocity Model assumes that the body pose at time t+1 will be the same as the velocity at time t. This means the pose is expected to change at a constant rate and direction.
  - It predicts the future pose by assuming the body continues to move as it was in the previous time step.
  - When evaluated the accuracy of the model, we received a MPJPE error of approximately 75 mm for a 1 second prediction

```
Constant Position MPJPE: 59.97 mm
Constant Velocity MPJPE: 75.67 mm
Improvement: -26.19%
```

PennState

Constant Position and Constant Velocity for an output sequence of 1 frame

- Our constant velocity model was outputting slightly higher MPJPE error than our constant position model for an output of 30 frames.
- We calculated the MPJPE for a shorter horizon of just 1 frame
  - For constant position, we got an MPJPE of 7 mm
  - For constant velocity, we got an MPJPE of 4 mm
- Small deviations from the constant velocity assumption accumulate over time. If the model is off by a little bit in the first few predicted frames, that error gets compounded in subsequent predictions, leading to a large error by the 30th frame.

```
Evaluating Constant Position Model (output_len=1):
Constant Position MPJPE: 7.79 mm

Evaluating Constant Velocity Model (output_len=1):
Constant Velocity MPJPE: 4.03 mm
```

GRU model architecture

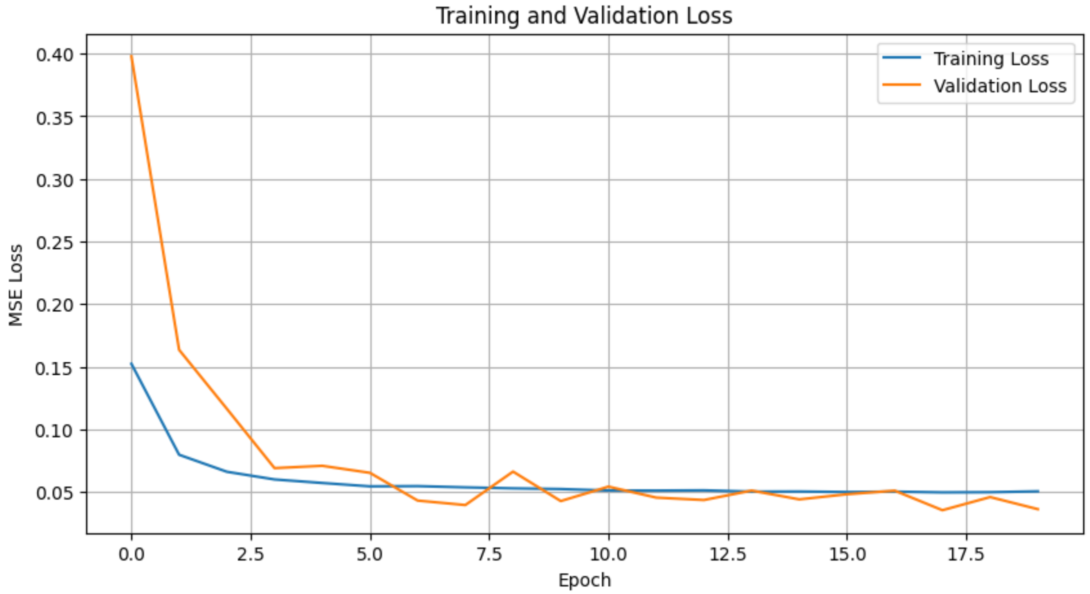
- We used a MotionGRU model to train our data. MotionGRU is a type of recurrent neural network specifically designed to predict human motion. RNNs are good at processing sequential data, like motion capture sequences.
- **Key components:**
  - **Encoder:** The encoder GRU takes the input motion sequence (past poses) and compresses it into a meaningful internal representation (a hidden state).
  - **Decoder:** The decoder GRU takes the encoded representation and generates the predicted future motion sequence (future poses).
  - **Projection:** The projection layers convert the decoder's hidden state into a pose representation that can be used by the body model.
  - **Teacher Forcing:** A technique to improve training, where the decoder sometimes uses the *actual* previous pose instead of its *predicted* previous pose.

PennState

Training

- We trained our model for 20 epochs on our dataset.
- We used MSE as the error measure for the training, learning rate of 0.0005, and the optimizer as the Adam optimizer.

```
Epoch 1/20, Train Loss: 0.1524, Val MPJPE: 53.55 mm
Model saved with validation MPJPE: 53.55 mm
Epoch 2/20, Train Loss: 0.0797, Val MPJPE: 33.53 mm
Model saved with validation MPJPE: 33.53 mm
Epoch 3/20, Train Loss: 0.0661, Val MPJPE: 28.96 mm
Model saved with validation MPJPE: 28.96 mm
Epoch 4/20, Train Loss: 0.0600, Val MPJPE: 21.88 mm
Model saved with validation MPJPE: 21.88 mm
Epoch 5/20, Train Loss: 0.0572, Val MPJPE: 21.56 mm
Model saved with validation MPJPE: 21.56 mm
Epoch 6/20, Train Loss: 0.0545, Val MPJPE: 21.79 mm
Epoch 7/20, Train Loss: 0.0547, Val MPJPE: 16.65 mm
Model saved with validation MPJPE: 16.65 mm
Epoch 8/20, Train Loss: 0.0537, Val MPJPE: 16.71 mm
Epoch 9/20, Train Loss: 0.0529, Val MPJPE: 20.43 mm
Epoch 10/20, Train Loss: 0.0524, Val MPJPE: 16.84 mm
Epoch 11/20, Train Loss: 0.0512, Val MPJPE: 19.78 mm
Epoch 12/20, Train Loss: 0.0511, Val MPJPE: 17.27 mm
Epoch 13/20, Train Loss: 0.0513, Val MPJPE: 17.12 mm
Epoch 14/20, Train Loss: 0.0503, Val MPJPE: 17.50 mm
Epoch 15/20, Train Loss: 0.0504, Val MPJPE: 16.80 mm
Epoch 16/20, Train Loss: 0.0500, Val MPJPE: 17.90 mm
Epoch 17/20, Train Loss: 0.0503, Val MPJPE: 17.93 mm
Epoch 18/20, Train Loss: 0.0497, Val MPJPE: 14.70 mm
Model saved with validation MPJPE: 14.70 mm
Epoch 19/20, Train Loss: 0.0498, Val MPJPE: 18.37 mm
Epoch 20/20, Train Loss: 0.0505, Val MPJPE: 14.94 mm
Motion GRU MPJPE on Test Set: 115.22 mm
```



PennState

Difficulties

- The trained GRU model shows promising loss curves but fails to produce valid body meshes
- Key Suspects:
    - If the latent vectors or predictions are not properly scaled before decoding, the body meshes may be out of proportion, resulting in invalid meshes.
    - Our data pipeline might not be handling the temporal dependencies between the data correctly which did not lead to the right output.
    - We could have trained the model for a higher number of epochs.

Future work

- With some more time, GPU capacity, and a deeper understanding of the project, we would like to investigate why we are not able to generate the visualizations from the weights we trained our GRU model with.

PennState

Thank you.

PennState