Computer Animation and Special Effect
Group 32

110612117 張仲瑜, 110550022 賴柏允

- Goal

To implement Mofusion, a diffusion-based motion synthesis model that can generate a sequence of motion based on the given text prompt.

- Implementation Details

We chose to implement the losses and the time-varying weight schedule from MoFusion on top of an older paper, MotionDiffuse. MotionDiffuse trains the model on local bone positions and rotations totaling to 263 features and only uses the mean squared error between the real noise added to the original motion snippet and the predicted noise from the model.

The main difference between MoFusion and MotionDiffuse is that MoFusion trains the model on global positions rather than the usual local joint positions and rotations.
We do this by converting the joint positions and rotations to global coordinates with forward kinematics, resulting in 22 joints each with their x, y and z coordinates for a total of 66 features.

The authors also proposed using 3 additional losses to train the network on top of the MSE Loss (In the MoFusion paper, it was stated to be L2 distance, but due to the difference in motion data length, we feel that MSE should be a better estimate).
These 3 are all skeletal losses, so it is needed to form a prediction from the generated noise. This is done by using the re-parameterisation trick commonly used in diffusion models:

$$\hat{\mathbf{M}}^{(0)} = \frac{1}{\sqrt{\bar{\alpha}_t}}\mathbf{M}^{(t)} - \left(\sqrt{\frac{1}{\bar{\alpha}} - 1}\right) f_\theta(\mathbf{M}^{(t)}, t, c).$$

However, the authors argue that naively using this formula to approximate the reverse-diffusion outputs leads to unstable training because the

generated motion is extremely noisy when near the end of the timesteps, or t=T.
So they remedy this by multiplying all the skeletal losses by a weight
λ (t) k = α_bar_t
where alpha bar is the cumulative product of all alphas (1 - noise schedule. The higher the noise schedule, the more noise is added to the original motion) up to time step t, so the noisier motion will contribute less to the overall loss.

The first skeletal loss mentioned minimizes the temporal variance of bone lengths. This ensures that the skeleton does not distort throughout the generated motion.

The second loss minimizes asymmetry between paired bones. This improves the anatomy of the generated skeleton to avoid deformation.

The final loss is also a ground truth supervision loss similar to the first MSE loss, but is instead done between the predicted and original motion.

One thing to note is that the losses should be inferred from masked tensors. The reason is that all motion data are padded with zeros to the same length before being loaded. After introducing noise, the padded sections do not contain any meaningful information and it will be detrimental to the overall accuracy of the model if those segments were accounted for in the loss calculations.

- Encountered difficulties

1. We forgot to implement masked mean and variance
2. Tensor operations are quite unintuitive
3. The original codebase calculated loss in two places but only one was used, causing our modifications to not go through back propagation initially.
4. Weak ass graphics card 🗿

- Demo video

https://youtu.be/PMwYrDMbZ-0

- Discussion

- Comparison with original method

1. The original method trains the model on 263 features while MoFusion trains only on 66, so the former would require more memory during training. On our machine with a single 3060ti, the increased memory consumption actually caused the original model to crash at the 35th epoch, while the modified model was able to finish all 50 epochs. The loading times during prediction are also faster.
2. As described in the MoFusion paper, a model trained on global position does seem to produce more reasonable results compared to one trained on rotations. The modified model sees less sliding when the given prompt contains motions that should be performed stationary, even without a foot-sliding loss.

- Time-varying weight schedule

From the results, we found out that the time-varying weight schedule plays a massive part in refining the results. Without it the model struggles to even learn bigger movements and the generated skeleton retains a certain amount of noise.

- Effectiveness of the losses

The 3 proposed losses in the MoFusion paper were enough to keep the skeletal structure intact, but we found various cases where certain bone pairs were abnormally long as the losses only ensured that bones have constant length and are symmetrical throughout the motion.

- Github repository

https://github.com/kwkwkwkak/CA-Final-Project

- Thoughts

The project is not as easy as we thought in the beginning. We’ve encountered many difficulties and even considered giving up. Thankfully, as we put in more time, patience, and effort to adjust and pivot, we boldly overcame most of the obstacles. In the end, our efforts paid off. It’s a valuable lesson for us to learn to persevere until the very end, not just until the last moment.