

Generative Motion Infilling From Imprecisely Timed Keyframes, Supplemental

1 Diffusion Model Hyperparameters

- Number of Transformer Layers: 8
- Latent Dimension Size: 512
- Number of Attention Heads: 4
- Feed-forward Dimension Size: 1024
- Dropout: 0.1
- Activation: gelu
- Noise Schedule: cosine
- Training Batch Size: 64
- Learning rate: 10^{-4}

2 Metrics

The goal of our quantitative evaluation is to study how well our model and baselines can reconstruct the *intended* motion when given *approximately timed* keyframe constraints. In the absence of a large dataset containing pairs of “intended motions” and their corresponding keyframes, we instead treat ground truth motions from the HumanML3D test set \mathbf{Y}_{true} as the “intended motion”, and automatically generate \mathbf{X} , in the same manner as our dataset generation method, as *plausible constraints* intended to generate \mathbf{Y}_{true} .

Reconstruction. Reconstruction accuracy to measures how well generated motions Y match the dynamics of the ground truth motion \mathbf{Y}_{true} . We

measure this by computing L2 distance between the global and local joint positions (L2-Pos), velocities (L2-Vel), accelerations (L2-Acc), and jerks (L2-Jerk) of the generated and ground truth motions. We include metrics on higher-order statistics, such as acceleration and jerk, because accurate timing reconstruction is noticeable in these measures, and they are more sensitive to timing discrepancies compared to just position or velocity.

Keypose Error. Because input constraint timings are permitted to change, we report keypose error (KPE) as the lowest L2-Pos between $\mathbf{X}_{k+\Delta k}$ (the ground truth keyframe that was temporally perturbed) and all poses in the generated \mathbf{Y} within 10 frames of k .

Diversity. Our diversity metric tests the ability of the model to produce different \mathbf{Y} , given the same \mathbf{X} , by measuring variability in generations with the same input. To test diversity, we sample 330 different \mathbf{X} at random, and run our model and baselines on each \mathbf{X} 64 times. Similar to prior work, we compute diversity by sampling two subsets of size 64 from the set of all \mathbf{Y} generations for the same \mathbf{X} , denoted $\{v_1, v_2, \dots, v_{64}\}$ and $\{v'_1, v'_2, \dots, v'_{64}\}$, and calculate diversity in local space as

$$\text{Diversity} = \frac{1}{64} \sum_{i=1}^{64} \|v_i - v'_i\|_2 \quad (1)$$

More information about this metric can be found in [Cohan et al.(2024), Guo et al.(2022)]

Jitter. We calculate jitter as the average of the third derivative of joint positions, in local space. Jit-

ter is a common quantitative metric for evaluating motion quality.

Note: In our evaluation, basic foot-skate cleanup is applied to the qualitative results in our supplemental videos; however, no postprocessing is applied to the outputs for any method during quantitative evaluation.

3 Quantitative Performance vs Δk

We train our model with $p = 5$, meaning that during dataset generation, keyframes are shifted by up to $|\Delta k| = 5$ frames. In this section, we analyze inference-time performance when keyframe shifts exceed this range.

Our experimental setup is similar to the quantitative experiments described in the main paper: given motion clips from the HumanML3D test set, we slice them into F -frame sequences and randomly select an extrema pose. The selected pose is then temporally shifted by $\pm\Delta k$, where Δk is selected uniformly at random between $[0, 25)$ frames. This range corresponds to no timing error ($\Delta k = 0$) up to a significant keyframe error of more than a second ($\Delta k = 25$). We consider the upper bound of $\Delta k = 25$ to be quite high and unlikely at test time, as we expect users to specify keyframe timing well within a second of the ground truth location. However, we choose this upper bound to evaluate the model’s robustness under extreme conditions.

We plot the following metrics in Fig. 1:

- Reconstruction accuracy, as measured by L2-Pos, L2-Vel, L2-Acc, and L2-Jerk in local space.
- KPE, the lowest L2-Pos between $\mathbf{X}_{k+\Delta k}$ (the ground truth keyframe that was temporally perturbed) and all poses in the generated \mathbf{Y} within 25 frames of k .
- Motion quality, as measured by two standard quantitative metrics of physical plausibility: jitter (average of the third derivative of joint po-

sitions, in local space) and foot-skate ratio (as described in [Guo et al.(2022)]).

KPE is lowest within the dataset range, then increases steadily as Δk grows outside of the dataset range This trend indicates that the model performs well at predicting keyframe locations when the temporal shifts are within the range it was trained on (i.e., $|\Delta k| \leq 5$). As Δk increases beyond this range, the error rises. This behavior is expected, as the model has no exposure to larger shifts during training.

Reconstruction accuracy decreases as Δk increases, but motion quality remains stable. This pattern suggests that larger keyframe shifts introduce greater ambiguity, making it more challenging for the model to reproduce the exact intended motion. Nevertheless, the model still generates a high-quality motion sequence (as measured by jitter and foot-skate metrics)—albeit a different one from the ground truth.

Reconstruction accuracy is highest when Δk is in the dataset range and decreases as Δk increases. Despite the increased reconstruction errors, motion quality metrics remain relatively stable across different values of Δk , even when the keyframe shifts are well outside the dataset range. This observation underscores the model’s robustness in generating plausible motions even when exact adherence to the ground truth sequence is not possible.

4 Visualizing Time-warp curves

We visualize some of the time warp curves predicted by our model. Despite the fact that smoothness of \mathbf{w} is not directly enforced in the objective (nor through any reconstruction loss on \mathbf{w} itself), we find that the predicted warp curves are globally quite smooth.

In Figure 3, we graph predicted time-warp curves from 100 generations of our model, given three keyframes of a character punching. There is a pose at frame 0 (start of the motion), at frame 15 (punching arm is fully outstretched), and frame 59 (end of the motion). We see that the time-warps are quite smooth. The model is also self-consistent, i.e., the

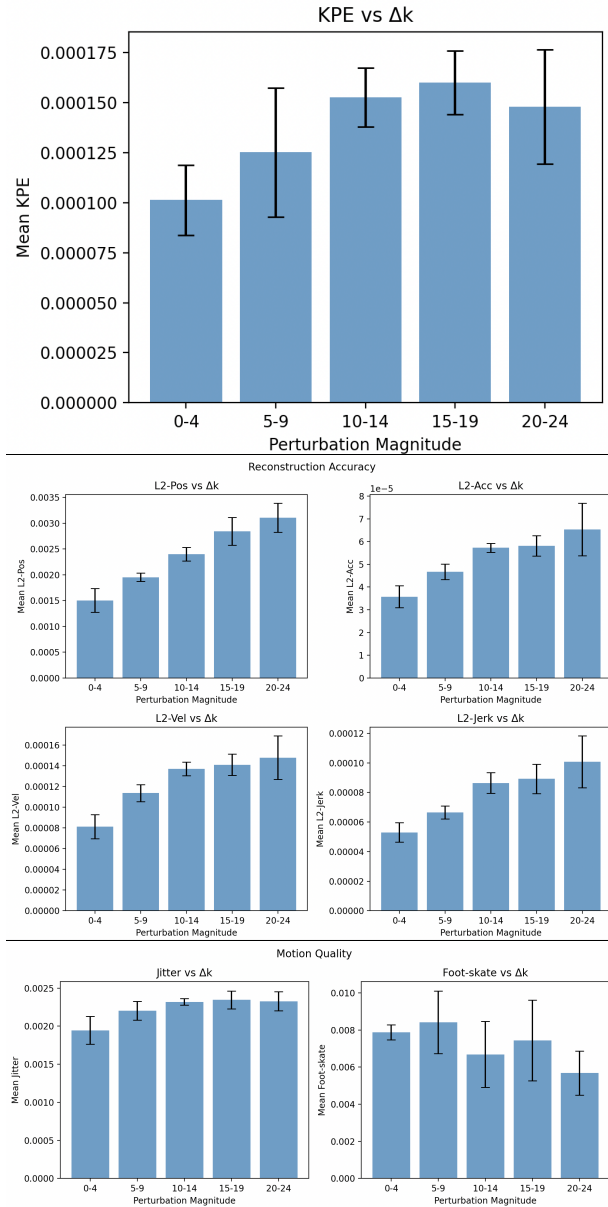


Figure 1: Bar graphs of KPE (top row), reconstruction accuracy (middle row), and motion quality metrics (bottom row) vs. $|\Delta k|$. On the x-axis, Δk is grouped into intervals of 5 frames.

time-warps do not display drastic deviations from each other, but have subtle differences to explore the timing design space for the given motion.

In Figure 4, we graph predicted time-warp curves from 100 generations of our model, from a motion editing scenario: an animator edits a motion of a character kicking to kick a second time (see *additional_results.mp4* timestamp 4:37) by inserting a new pose at frame 45. Time-warps are smooth and consistent, showing a trend for speeding up frames in the center of the motion; in the final motion, we see a much snappier kick around this time than appears in the blocking motion.

In Figure 5, we graph predicted time-warp curves from 100 generations of our model, from a motion synthesis scenario: an animator creates a motion where a character ducks, then kicks, then ducks (see *main_video.mp4* timestamp 0:41, and main paper, Figure 1). Time warps are smooth and show an interpretable effect: the motion is slowed down so that the first duck happens later in the motion.

5 Implementation of Learned Time Warp

We implement the time warp, which, as mentioned in the main paper, we parameterize as a backward mapping, using Pytorch’s `grid_sample` function. `grid_sample` is inherently a backward mapping operation, designed to sample values from the input tensor based on coordinates provided by a flow field grid. In our case, this grid represents a time warp, where each value in the grid indicates the corresponding source time index from which to sample.

By leveraging `grid_sample`, we ensure differentiability, allowing the gradients to propagate through the time warp operation, thus enabling learning of the time warp function itself. We use `grid_sample`’s built in bilinear interpolation functionality to ensure smooth transitions.

6 CondMDI and IMP Baseline

6.1 Re-training CondMDI and IMP

The original **CondMDI** is trained to handle motions of arbitrary lengths up to 196 frames and includes text conditioning. The same is true for the diffusion model used for **IMP**. For a more apples-to-apples comparison, we retrain **CondMDI** and **IMP** on the 60 frame motion-clips, spliced from the motions in the HumanML3D dataset (matching our own set up). We also remove text conditioning when training, so that the model is only conditioned on keyframes (as opposed to keyframes and text).

6.2 Performance Comparison: Re-Trained vs. Original CondMDI

In Table 1 we show the test set performance over all our reconstruction metrics of the re-trained **CondMDI** (shown as **CondMDI(60)** in the table), vs the original **CondMDI** (shown as **CondMDI(Original)** in the table). **CondMDI(Original)** is the pre-trained model from [Cohan et al.(2024)], which we run on our test set with an empty string for text conditioning. We find that **CondMDI(60)** is improved over **CondMDI(Original)** on most of our reconstruction metrics, but still does not match the results of our proposed model **LT**.

6.3 Trajectory drift in CondMDI generations

We choose **CondMDI** as the representative motion-inbetweening work to compare against, because it is the current state-of-the art in inbetweening, and is trained on the same dataset (HumanML3D). As described in the paper and evident in a small number of qualitative examples in the video, **CondMDI** occasionally generates motions with a global trajectory drift from the input keyframe constraints. We have also occasionally noticed a few motion quality artifacts that are mentioned in **CondMDI**’s limitation section, such as motion jitter. This occurs for both **CondMDI(60)** and **CondMDI(Original)**. For

all our results, qualitative and quantitative, using **CondMDI**, we use the official codebase provided by the authors. We have confirmed with the authors of the paper that we are using the codebase correctly, and that they have sometimes noticed similar issues. Our hypothesis for this drift is it may be caused by **CondMDI**’s motion representation, which includes a global root trajectory, but represents all other joints only in local space. It may be quite easy for the model to diverge in root trajectory but still get the local motion correct, because the global trajectory is a comparatively small part of the feature vector.

Regardless, our goal is to demonstrate that the *task* definition of learned motion-inbetweening is not suitable for the context where keyframes may be imprecisely timed. Our model has a significant quantitative performance increase over **CondMDI** even in local space, and qualitative performance increase even in cases where **CondMDI** does not demonstrate global trajectory drift .

7 Interpretation of the Learned Time-warp

To assess how effectively the globally learned function can retime the $\mathbf{X}_{k+\Delta k}$ to the *correct* frame location k , we conduct the following evaluation: for each ground-truth $\mathbf{X}_{k+\Delta k}$, we find the most similar pose \mathbf{Y}_r in the generated \mathbf{Y} set, and then calculate the frame distance between frame r and frame k . The closer this distance is to 0, the better \mathbf{Y}_r has been retimed to the correct location.

We present our results as a histogram in Fig. 2. As shown, **LT** has the highest number of instances where $|r - k| \leq 1$ (5237, vs **CondMDI** 3996, **NoTime** 4579, **NoWarp** 5052, **IMP(0)** 4389, **IMP(1)** 953, **IMP(5)** 820), and a smaller average $|r - k|$ value within $|r - k| \leq 5$ compared to the baselines (1.88, vs **CondMDI** 2.34, **NoTime** 2.21, **NoWarp** 2.00, **IMP(0)** 2.44, **IMP(1)** 2.66, **IMP(5)** 2.715). Having seen mistimed keyframes during training, **NoWarp** performs second-best, but without a mechanism to retime the keyframe, cannot score higher than **LT**, underscoring the importance

Table 1: **Metrics.** Comparing **CondMDI(60)** with **CondMDI(Original)**. **CondMDI(60)** is improved across almost all reconstruction metrics, but still does not beat our results (as in the main paper, we show our results as **LT** in the bottom row).

	L2-Pos (10^{-1}) G/L ↓	L2-Vel (10^{-4}) G/L ↓	L2-Acc (10^{-4}) G/L ↓	L2-Jerk (10^{-3}) G/L ↓	KPE (10^{-2}) ↓	Jitter (10^{-2}) ↓	Diversity ↑
CondMDI(60)	0.43 / 0.069	5.88 / 2.39	5.87 / 1.52	1.77 / 0.37	0.120	0.75	2.43
CondMDI(Original)	0.67 / 0.066	5.91 / 2.34	6.83 / 1.81	2.09 / 0.48	0.159	0.84	2.54
LT (Ours)	0.03 / 0.017	1.35 / 1.02	0.60 / 0.46	0.096 / 0.068	0.019	0.22	3.68

of the learned warp.

As discussed earlier, coordinate-based metrics are not always reliable due to the possibility of multiple plausible timings for a given input. $|r - k|$ also does not take account of motion quality. In the test set, keyframes are perturbed up to ± 5 frames, and **IMP(0)** directly overwrites these mistimed keyframes into the output. Therefore, all **IMP(0)** generations fall within $|r - k| \leq 5$, and it scores higher than **IMP(1)** and **IMP(5)**. But, as discussed in the main paper, **IMP(0)** produces lower-quality motion than other imputation-based baselines.

Ultimately, we consider $|r - k|$ as an aggregate, rough indication of correctness. Therefore, this metric should be considered alongside all the other quantitative metrics, and their effects should be assessed collectively. We also encourage readers to interpret this metric in the context of our qualitative results.

7.0.1 Difference Between Ground-truth w and Predicted w

Recall that in our data generation process, we temporally shift the keyframe at frame k by Δk and delete a window W of frames around $k + \Delta k$ in range $(k - W, k + \Delta k)$ and $(k + \Delta k + 1, k + W)$. A ground truth time-warp \mathbf{w}_{gt} can be defined for this perturbation. For frames below $k - W$ and above $k + W$, $\mathbf{w}_{gt}(t) = t$; original time index k is mapped to to $k + \Delta k$. The predicted timewarp \mathbf{w} represents the model’s estimation of this mapping. To evaluate the difference between the ground truth and predicted time-warps, we compute the mean squared error between \mathbf{w}_{gt} and \mathbf{w} over **LT** generations. This metric provides a measure of the deviation between the pre-

dicted and actual warp; lower values indicate better alignment. We find the warp error to be 0.0088.

8 Replicability

We will release all model weights, source code, and metadata.

References

- [Cohan et al.(2024)] Setareh Cohan, Guy Tevet, Daniele Reda, Xue Bin Peng, and Michiel van de Panne. 2024. Flexible Motion In-betweening with Diffusion Models. arXiv:2405.11126 [cs.CV] <https://arxiv.org/abs/2405.11126>
- [Guo et al.(2022)] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. 2022. Generating Diverse and Natural 3D Human Motions from Text. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5142–5151. <https://doi.org/10.1109/CVPR52688.2022.00509>

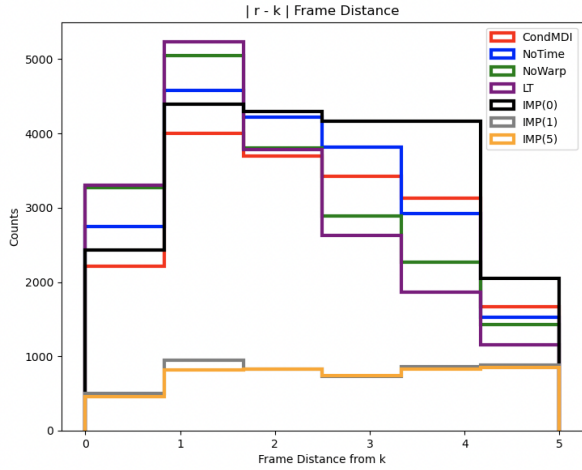


Figure 2: Histogram of $|r - k|$ frame distance between the ground truth keyframe location k and the frame r containing the pose that most closely matches $\mathbf{X}_{k+\Delta k}$. The graph visualizes results where $|r - k| \leq 5$. Notice how **LT** (purple) produces motions where the most $|r - k| \leq 1$, the least motions where $|r - k|$ is high, and the lowest average $|r - k|$ (1.88). This suggests that **LT** can on average retime \mathbf{X} more correctly than baselines, highlighting the importance of the learned warp for accurately retiming motions. Since keyframes in the test set are perturbed by up to ± 5 frames, **IMP(0)** simply replicates these mistimed keyframes in its output, ensuring all its results fall within $|r - k| < 5$ (notice its high density and roughly even distribution within this range).

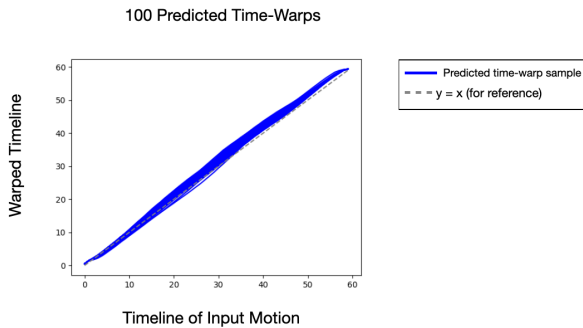


Figure 3: 100 predicted time-warps for an input keyframes of a character punching.

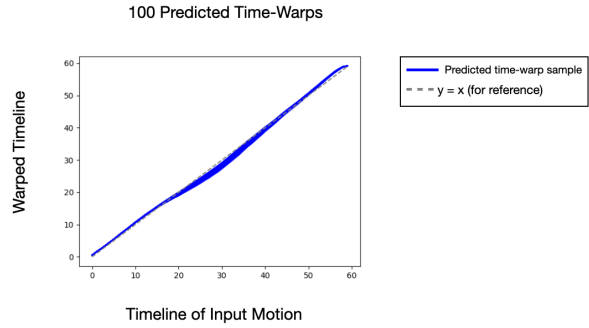


Figure 4: 100 predicted time-warps generated from \mathbf{X} that edits an existing sequence of a character kicking, to a character kicking twice .

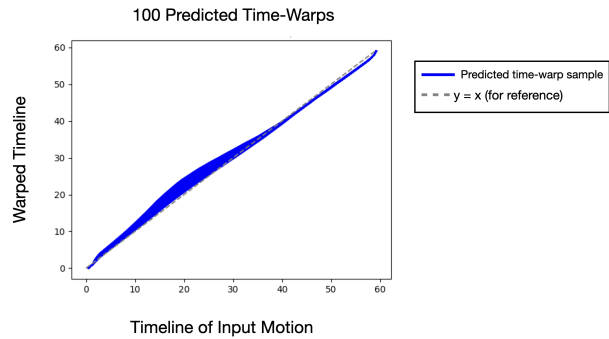


Figure 5: 100 predicted time-warps generated from \mathbf{X} that creates a motion where a character ducks, then kicks, then ducks .