

# A Unified 3D Human Motion Synthesis Model via Conditional Variational Auto-Encoder

## 1. Derivation of the Conditional Variational Auto-Encoder (CVAE)

Given the observed regions  $\mathbf{X}_i$  in a pose series, we attempt to synthesize a plausible pose sequence  $\hat{\mathbf{X}}_g = \{\mathbf{X}_i, \hat{\mathbf{X}}_u\}$  without losing the prior knowledge of the input key frames. Mathematically, our goal is to maximize the posterior probability  $p(\mathbf{X}_u|\mathbf{X}_i)$

To this end, we resort to a CVAE [8] (conditional variational auto-encoder) based framework, which estimates a parametric distribution  $p_\phi(\mathbf{z}_u|\mathbf{X}_i)$  of the unseen regions over a latent space, from which to sample the latent vector  $\mathbf{z}_u$  and further generate the full sequence  $\hat{\mathbf{X}}_g$ . Similar to the original derivation, we start with the posterior formulation that we wish to maximize:

$$p(\mathbf{X}_u|\mathbf{X}_i) = \int p_\theta(\mathbf{X}_u|\mathbf{X}_i, \mathbf{z}_u)p_\phi(\mathbf{z}_u|\mathbf{X}_i)d\mathbf{z}_u, \quad (1)$$

which can be re-written as follows:

$$\begin{aligned} p(\mathbf{X}_u|\mathbf{X}_i) &= \int p_\theta(\mathbf{X}_u|\mathbf{X}_i, \mathbf{z}_u) \frac{p_\phi(\mathbf{z}_u|\mathbf{X}_i)}{q_\psi(\mathbf{z}_u|\mathbf{X}_u)} q_\psi(\mathbf{z}_u|\mathbf{X}_u) d\mathbf{z}_u \\ &= E_{\mathbf{z}_u \sim q_\psi(\mathbf{z}_u|\mathbf{X}_u)} [p_\theta(\mathbf{X}_u|\mathbf{X}_i, \mathbf{z}_u) \frac{p_\phi(\mathbf{z}_u|\mathbf{X}_i)}{q_\psi(\mathbf{z}_u|\mathbf{X}_u)}]. \end{aligned} \quad (2)$$

After taking logs and applying Jensen’s inequality, we obtain the variational lower bound of the conditional log-likelihood of the observation:

$$\begin{aligned} \log(p(\mathbf{X}_u|\mathbf{X}_i)) &\geq E_{\mathbf{z}_u \sim q_\psi(\mathbf{z}_u|\mathbf{X}_u)} [\log(p_\theta(\mathbf{X}_u|\mathbf{X}_i, \mathbf{z}_u)) \\ &\quad + \log(\frac{p_\phi(\mathbf{z}_u|\mathbf{X}_i)}{q_\psi(\mathbf{z}_u|\mathbf{X}_u)})] \\ &= E_{\mathbf{z}_u \sim q_\psi(\mathbf{z}_u|\mathbf{X}_u)} [\log(p_\theta(\mathbf{X}_u|\mathbf{X}_i, \mathbf{z}_u))] \\ &\quad - \mathbf{KL}(q_\psi(\mathbf{z}_u|\mathbf{X}_u) || p_\phi(\mathbf{z}_u|\mathbf{X}_i)). \end{aligned} \quad (3)$$

where  $\mathbf{KL}$  is the Kullback-Leibler divergence,  $\mathbf{z}_u$  the sampled latent vector,  $q_\psi(\mathbf{z}_u|\mathbf{X}_u)$  the posterior sampling function,  $p_\phi(\mathbf{z}_u|\mathbf{X}_i)$  the conditional prior, and  $p_\theta(\mathbf{X}_u|\mathbf{z}_u)$  is the likelihood. The conditional probability distributions  $q_\psi, p_\phi, p_\theta$  can be parameterized by deep neural networks.

## 2. Implementation Details

Our network is implemented in Pytorch framework. Following [10, 11], we applied spectral normalization [6] to stabilize the adversarial training. All networks are initialized with Orthogonal Initialization [7] and trained from scratch with a fixed learning rate of  $10^{-4}$ . We used the Adam optimizer with  $\beta_1 = 0$  and  $\beta_2 = 0.99$ . The appearance match loss in Equation 6 and Equation 7 of our main paper is used in four output scales, while the discriminator is only added after the final output series. For data-processing, the input and generated results are represented in 3d joint positions. The masked areas are set with zero values. The 3d joint positions are root-relative without removing global orientation. We trained each model on a single GTX V100 GPU, with a batch size of 128.

One may also be curious about how to variable length sequences. A straightforward way to handle variable pose series is to train the model with different lengths  $T$ . However, this may not be optimal when we wish to process different length

series with the same model. To this end, we assume two scenarios. If the input length  $\tau \leq T$ , we can directly take the first  $\tau$  frames as the generation result. If  $\tau > T$ , we suggest to first generate  $T$  frames and iteratively take a small number of the previously generated frames as input constraints to produce future motions.

### 3. Detailed Network Architecture

Our network architecture is inspired by SA-GAN [9] and BigGAN [1] and PICNet [10]. Details of our basic blocks and all of the network modules (including encoder, decoder, discriminator and action classifier) can be found in Figure 2 and Figure 3.

## 4. Extensive Experiments

### 4.1. Two branch CVAE vs Single branch CVAE.

We found that using one-branch CVAE with both reconstruction and generative constraints led to limited diversity of the generated series, where the diversity score of “ours w/o action” on Human3.6 dropped from 0.26 (two-branch CVAE) to 0.16 (one-branch CVAE). This is because the network tends to focus more on reconstructing the single solution ground truth. For two-branch CVAE, the upper branch focuses on generating plausible pose series, while the lower branch targets at accurate reconstruction, thus performing better in generating both diverse and high-quality results.

## 5. Qualitative Evaluation

### 5.1. Generation Diversity

We qualitatively evaluate the generation diversity of our proposed method in Figure 4. As can be seen, given the same input constraints, our model is able to produce diverse and plausible results by sampling multiple times from the latent distribution.

### 5.2. More Visualization Results

We provide more visual examples of our proposed method for various tasks in Figure 5. To gain more insight into the quality of the generated series, we present our results in attached videos. As shown in these videos, our approach can not only address different motion-based tasks with coherent and realistic results, but also manipulate the animation styles of the synthesized series according to the given action labels.

### 5.3. Additional Results on AMASS dataset

To show the generalization ability of our proposed method, we also provide qualitative results on AMASS dataset [4], we follow the train/test split of [5] to process data. The results can be found in Figure 1, showing that our proposed method is capable of generating plausible results on different datasets.



Figure 1. Qualitative examples of our proposed model on AMASS dataset. Gray poses are the pre-defined input frames (or partial body of some frames), while the red & blue skeleton sequences are the synthesized poses.

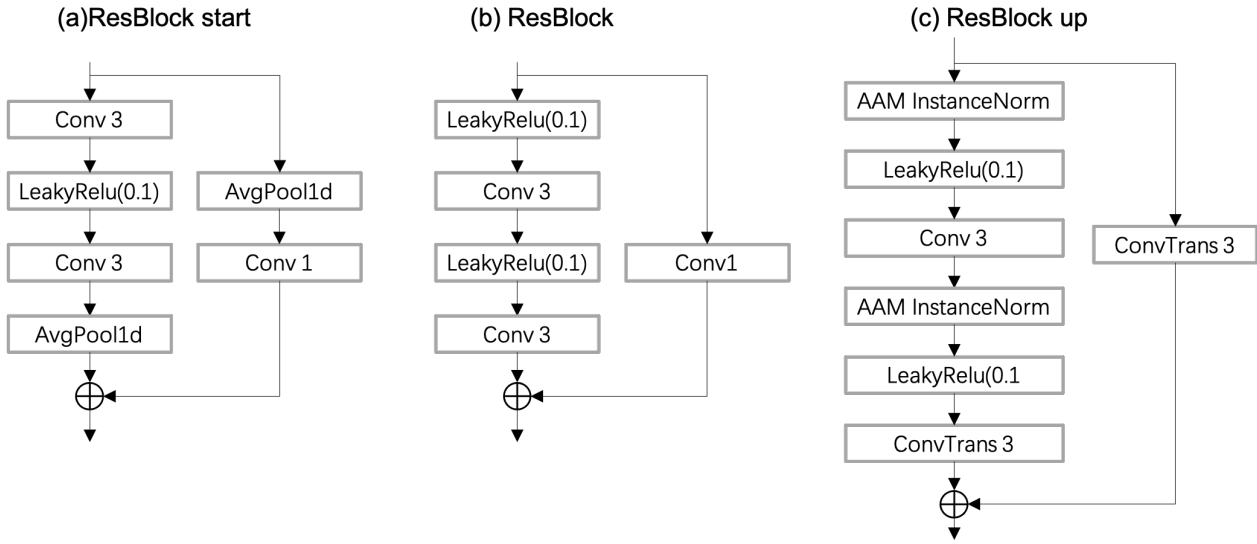


Figure 2. Illustration of the Basic Residual Blocks used in our model. (a) The start Residual Block for the encoder and discriminator networks. (b) The Residual Block used in all networks. (c) The Residual Block up used in the decoder (generator) network. The ResBlock downsample in Figure 3 means adding an average pooling layer after the Residual Block (b).

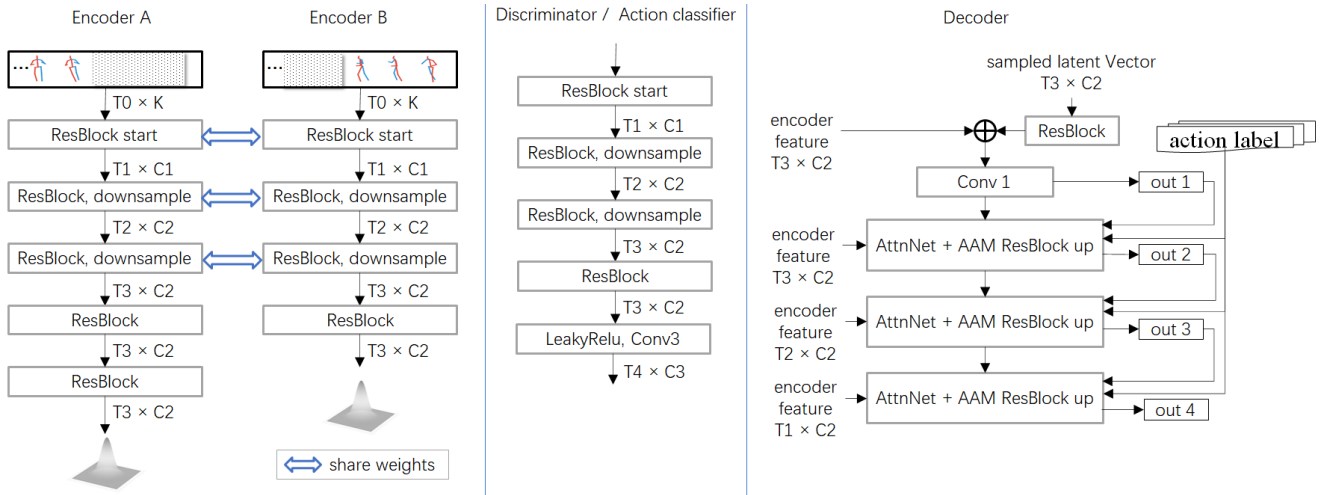


Figure 3. Architectures for our frameworks, including encoder, decoder, discriminator, and action classifier. For the discriminator, the input is the concatenation of generated sequence and action labels. For the action classifier, the input is the generated sequence.  $T_0 = 128$ ,  $T_1 = 64$ ,  $T_2 = 32$ ,  $T_3 = 16$ ,  $T_4 = 15$ ,  $C_1 = 128$ ,  $C_2 = 256$ ,  $C_3 = 1$  for discriminator and the number of classes for action classification network.  $K$  is the number of parameters describing each pose.

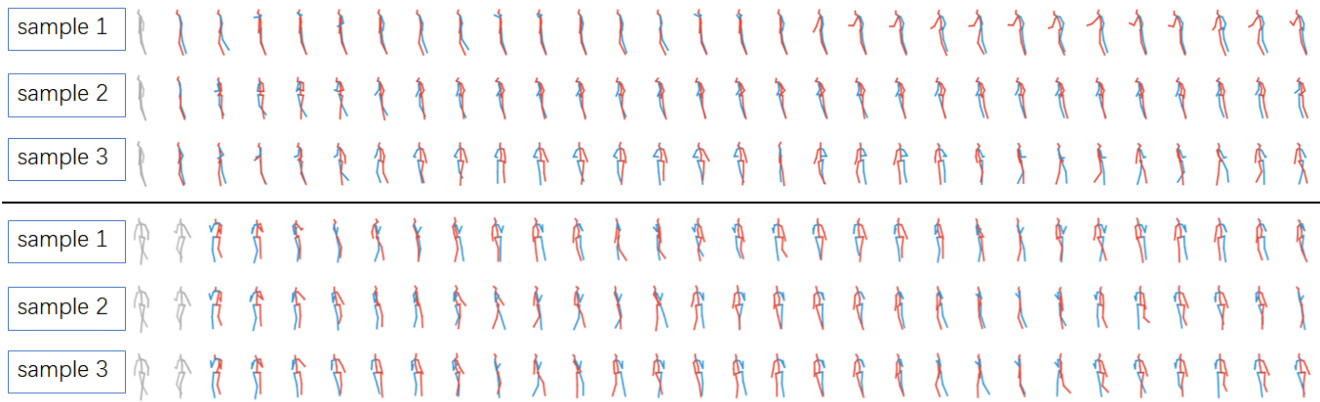
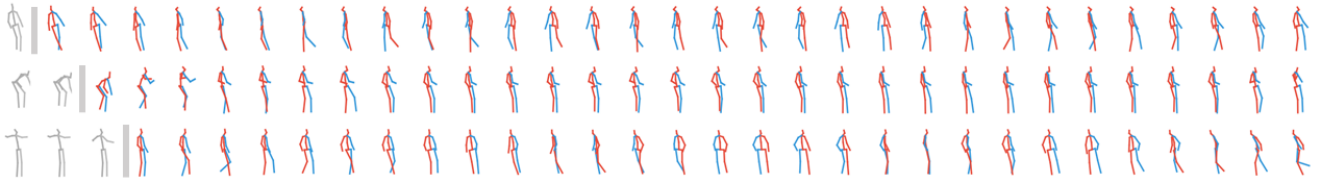
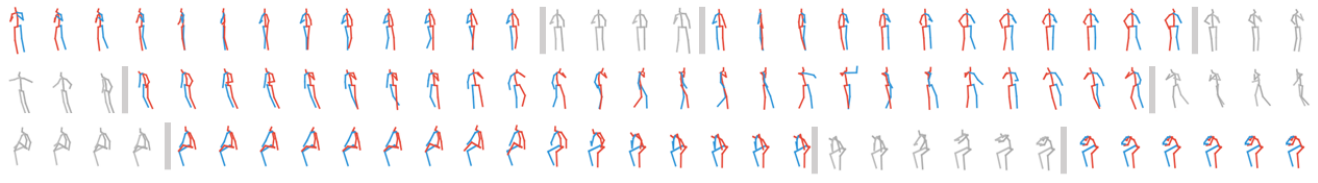


Figure 4. Qualitative analysis for the diversity of the generated pose series with the same input constraints. As can be seen, our model is able to generate diverse and plausible results given the same input conditions.

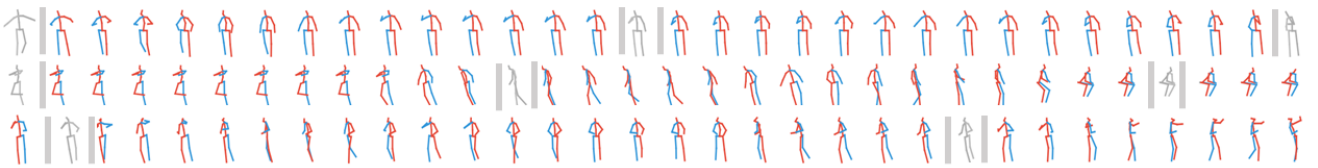
### Future prediction



### Completion



### Interpolation



### Spatial-Temporal Completion

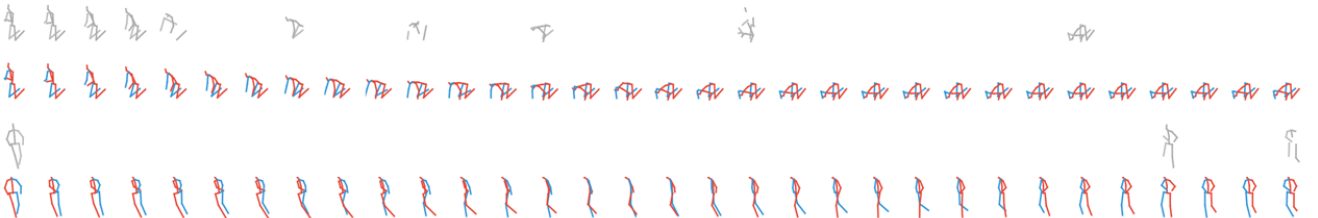


Figure 5. More qualitative examples of our proposed model for different motion-related tasks. Gray poses are the pre-defined input frames (or partial body of some frames), while the red & blue skeleton sequences are the synthesized poses. The input constraints can be flexibly set to arbitrary positions with varying densities.

## References

- [1] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.
- [2] Yujun Cai, Lihao Ge, Jun Liu, Jianfei Cai, Tat-Jen Cham, Junsong Yuan, and Nadia Magnenat Thalmann. Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2272–2281, 2019.
- [3] Haoshu Fang, Yuanlu Xu, Wenguan Wang, Xiaobai Liu, and Song-Chun Zhu. Learning pose grammar to encode human body configuration for 3d pose estimation. *arXiv preprint arXiv:1710.06513*, 2017.
- [4] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *International Conference on Computer Vision*, pages 5442–5451, Oct. 2019.
- [5] Wei Mao, Miaomiao Liu, and Mathieu Salzmann. History repeats itself: Human motion prediction via motion attention. In *European Conference on Computer Vision*, pages 474–489. Springer, 2020.
- [6] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.
- [7] Andrew M Saxe, James L McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*, 2013.
- [8] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *Advances in neural information processing systems*, pages 3483–3491, 2015.
- [9] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *International Conference on Machine Learning*, pages 7354–7363. PMLR, 2019.
- [10] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. Pluralistic image completion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1438–1447, 2019.
- [11] Peihao Zhu, Rameen Abdal, Yipeng Qin, and Peter Wonka. Sean: Image synthesis with semantic region-adaptive normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5104–5113, 2020.