Weakly-supervised Action Transition Learning for Stochastic Human Motion Prediction

Wei Mao[1], Miaomiao Liu[1], Mathieu Salzmann[2,3]
[1]Australian National University; [2]CVLab, EPFL; [3]ClearSpace, Switzerland
{wei.mao, miaomiao.liu}@anu.edu.au, mathieu.salzmann@epfl.ch

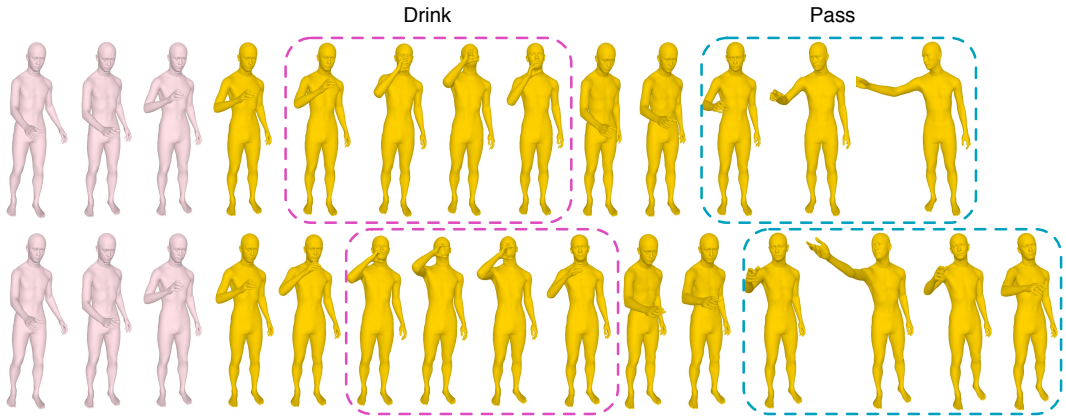


Figure 1. **Action-driven stochastic human motion prediction.** Given a past motion (pink) and a sequence of future action labels, our model generates action-specific future poses (yellow). We show two different futures generated for the same history and actions. Our model allows these predictions to have different lengths. The motions are down-sampled to the same frame rate for visualisation.

Abstract

We introduce the task of action-driven stochastic human motion prediction*, which aims to predict multiple plausible future motions given a sequence of action labels and a short motion history. This differs from existing works, which predict motions that either do not respect any specific action category, or follow a single action label. In particular, addressing this task requires tackling two challenges: The transitions between the different actions must be smooth; the length of the predicted motion depends on the action sequence and varies significantly across samples. As we cannot realistically expect training data to cover sufficiently diverse action transitions and motion lengths, we propose an effective training strategy consisting of combining multiple motions from different actions and introducing a weak form of supervision to encourage smooth transitions. We then design a VAE-based model conditioned on both the observed motion and the action label sequence, allowing us to generate multiple plausible future motions of varying length. We illustrate the generality of our approach by exploring its use with two different temporal encoding models, namely RNNs and Transformers. Our approach outperforms baseline models constructed by adapting state-of-the-art single action-conditioned motion generation methods and stochastic human motion prediction approaches to our new task of action-driven stochastic motion prediction. Our code is available at* https://github.com/wei-mao-2019/WAT.

1. Introduction

Modeling human motion has broad applications in human-robot interaction [26], virtual/augmented reality (AR/VR) [46] and animation [49]. As such, it has been an active research problem for many years [8]. In particular, recently, great progress has been made in predicting future motion given an observed past motion sequence [5,56]. Addressing this could have a significant impact on autonomous systems, allowing them to forecast potential dangers and plan their actions accordingly. Nevertheless, except for a few early methods that predict motions of a single action category [13, 21], recent methods [36, 56] mostly focus on action-agnostic predictions. Thus, they cannot be used by an autonomous system to generate specific potential future scenarios encoded by a sequence of action labels, for example to evaluate the consequences of a person on a sidewalk either *walking* to the crossing, *waiting* for the green light, and *crossing* the road, or instead *running* on the street and *stopping* in front of the car. By contrast, recent works on human motion synthesis can generate action-specific sequences [17, 42]. However, these methods neither lever-

age past motion observations, nor synthesize transitions between different actions. In this work, we therefore introduce the task of *action-driven stochastic human motion prediction*, which aims to predict a set of future motions given a sequence of action labels and past motion observations.

One of the key challenges of this task arises from the fact that humans can perform motions with all kinds of action transitions. For example, when one *walks* to a table, they can then either *grab* a drink, or *sit* on a chair, or *place* something on the table, or perform any combination of the above. Constructing a dataset that covers this huge space of possible action transitions is therefore virtually impossible, significantly complicating training a model for this task. As a matter of fact, to the best of our knowledge, almost all human motion datasets contain sequences that depict a single action. While the recent BABEL dataset [43] constitutes the only exception with multiple actions per sequence, it contains only a small subset of action transitions, which, as evidenced by our experiments, does not suffice to learn to generalize to arbitrary ones.

To tackle the diversity of human action transitions with such limited data, we develop a weakly-supervised training strategy that only relies on a motion smoothness prior. Specifically, we generate multi-action sequences by combining historical and future motions from different action categories, and account for the lack of supervision during the transition between two actions by simply encouraging the predicted motion to be temporally smooth. As will be shown by our experiments, such a simple prior suffices to model natural action transitions.

The second main challenge of our task arises from the stochastic nature of human motion: Several ways to perform one action sequence are equally plausible. To handle this stochasticity, we design a model based on a variational autoencoder (VAE) [24], conditioning the VAE on the observed past motion and on the action label sequence. We demonstrate the generality of this model by exploiting it with two different temporal encoding architectures, an RNN-based one and a Transformer-based one. Furthermore, to reflect the fact that some action sequences require more time to be executed than others, we introduce a simple yet effective strategy based on the prediction variance to produce multi-action motions of different lengths. This contrasts with most of the motion prediction literature, which predicts fixed-length motions, and, as illustrated by Fig. 1, allows us to generate realistic, diverse future motions depicting the given actions in order and with varying length. We believe that our approach can also be beneficial for other tasks, e.g., music generation of variable length.

Our contribution can therefore be summarized as follows: (i) We introduce a new task, *action-driven stochastic human motion prediction*, which bridges the gap between motion synthesis and stochastic human motion prediction; (ii) We propose a weakly-supervised training strategy to learn the action transitions without requiring an unrealistic amount of annotated data; (iii) We develop a simple yet effective way of predicting motions of varying length.

Our experiments on 3 human motion modeling benchmarks demonstrate the effectiveness of our approach, outperforming baseline models constructed by extending state-of-the-art action-conditioned motion synthesis methods and stochastic human motion prediction ones to our new task.

2. Related Work

Human Motion Prediction. Most human motion prediction works [4, 9, 10, 13, 15, 16, 21, 29, 36–38, 41, 52] focus on predicting human movements in a very short future (< 0.5s). These methods mainly differ in their temporal encoding strategies, using either recurrent architectures [13, 15, 16, 21, 38, 41, 52] or feed-forward models [4, 9, 10, 29, 36, 37]. Most of them, however, do not aim to produce motions with the same past sequence that respect any given action label. The only exceptions that incorporate action information are the early works of [13, 15, 16, 21, 38]. However, such information is only used to help predict future motion of the *same* action as the historical motion. By contrast, we seek to predict future motions of *different and multiple* action categories.

Because, given one sequence of action labels, we aim to predict multiple plausible motions, which is more closely related to diverse human motion prediction [5, 6, 18, 27, 32, 51, 54, 56]. To capture the distribution of future motions, these methods usually rely on deep generative models, such as VAEs [24] and generative adversarial networks (GANs) [14]. Among the most recent ones, the work of [5] prevents the VAE from ignoring the random variables by perturbing them; DLow [56] focuses on learning the sampling process for diverse future predictions from a pre-trained generative model. While these models indeed produce diverse and plausible future motions, these generated motions do not follow any clear semantic categories. As such, they could not be leveraged to help an autonomous system evaluate different scenarios defined in terms of semantic human behaviors. By contrast, our goal is to control the predicted future motion type using semantic information, i.e., a sequence of action labels. We therefore design a VAE-based model and a weakly-supervised training strategy that let us produce *different* plausible future motions for the *same* past motion and a sequence of action labels.

Human Motion Synthesis. In contrast to motion prediction, human motion synthesis aims to generate realistic human motions without any historical observations. While earlier works [39, 48] focused on simple, cyclic movements, e.g., using Principal Component Analysis [39] or the Gaussian process latent variable model [48], recent deep-learning-based methods [2, 17, 28, 30–32, 42, 45] can handle

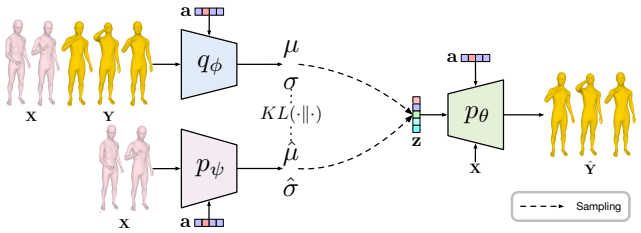


Figure 2. **Overview of our approach.**

more complicated motions. In this context, several works have proposed to condition the generated motions on some auxiliary signal, such as audio/music [28, 30, 31, 45] or textual descriptions [2, 32].

The methods most closely related to ours are [17, 42] which aim to generate action-specific human motions. In particular, [17] introduces a frame-level VAE-based model conditioned on the action label, and [42] a Transformer [50]-based VAE model with a sequence-level latent embedding. However, these models can only generate motions depicting individual actions. In principle, given supervised data covering all possible action transitions, they could be trained to generate more complex motions. However, such data cannot practically be obtained. We overcome this by designing a weakly-supervised training strategy that lets us leverage limited, single-action sequences.

Variable-length Motion Prediction. Although generating sequences of variable length has been well-studied for machine translation [50], it is rarely considered in human motion prediction/synthesis. However, as studied in [1] in the context of predicting future actions' semantics and duration from video data, different action categories, or even instances of the same action, vary significantly in length. Our method produces a variable-length future motion given an action label and a past motion. While [42] also generates motions of variable lengths, these length must be set manually. In contrast, we automatically find the appropriate duration by learning the distribution of motion lengths.

Action-conditioned Video Generation. The work most closely related to ours is PSGAN [55], which aims to predict future 2D human poses given one input image and a target action label. However, with only one input image, PSGAN cannot predict action transitions. Other action-conditioned generative methods include [53] and [22]. However, these works aim to generate face images conditioned on emotions, and the next game screen conditioned on keyboard actions, respectively, which both fundamentally differ from our task.

3. Our Approach

Let us now introduce our approach to action-driven stochastic human motion prediction. To represent a human in 3D, we adopt the SMPL model [34], which parametrizes a 3D human mesh in terms of shape and pose. Since we focus on human motion and not human identity, our model follows that of ACTOR [42] to only predict the pose parameters. The shape parameters are used for visualization only. Given an action label represented by a one-hot vector $\mathbf{a}$ and a sequence of N past human poses represented by $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N] \in \mathbb{R}^{K \times N}$, where $\mathbf{x}_i \in \mathbb{R}^K$ is the pose in the i-th frame, our goal is to predict a future motion $\hat{\mathbf{Y}} = [\hat{\mathbf{y}}_1, \hat{\mathbf{y}}_2, \cdots, \hat{\mathbf{y}}_T] \in \mathbb{R}^{K \times T}$, with $\hat{\mathbf{y}}_i \in \mathbb{R}^K$, representative of the given action label. To learn to predict transitions between different actions, as discussed in more detail below, we will train our model using data where $\mathbf{X}$ and the corresponding ground-truth future motion $\mathbf{Y}$ depict different actions. This will eventually allow us to predict future motions for sequences of action labels, by recursively treating the previous prediction as historical information.

3.1. Action-driven Stochastic Motion Prediction

To predict action-driven future motions, we design a model based on conditional VAEs (CVAEs) [24], whose goal is to model the conditional distribution $p(\mathbf{Y}|\mathbf{X}, \mathbf{a})$. Specifically, as shown in Fig. 2, we first model the posterior distribution $q_\phi(\mathbf{z}|\mathbf{Y}, \mathbf{X}, \mathbf{a})$ via a neural network, the encoder, where $\mathbf{z}$ is a latent random variable, and ϕ denotes the parameters of the encoder. From the latent variable $\mathbf{z}$, the CVAE then aims to reconstruct the future motion $\mathbf{Y}$ using another neural network, the decoder, expressed as $p_\theta(\mathbf{Y}|\mathbf{z}, \mathbf{X}, \mathbf{a})$, with parameters θ. The evidence lower bound (ELBO) of the conditional distribution $p(\mathbf{Y}|\mathbf{X}, \mathbf{a})$ can then be written as

$$\log p(\mathbf{Y}|\mathbf{X}, \mathbf{a}) \geq \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{Y}, \mathbf{X}, \mathbf{a})}[\log p_\theta(\mathbf{Y}|\mathbf{z}, \mathbf{X}, \mathbf{a})] - KL(q_\phi(\mathbf{z}|\mathbf{Y}, \mathbf{X}, \mathbf{a}) \| p_\psi(\mathbf{z}|\mathbf{X}, \mathbf{a})) \,, \quad (1)$$

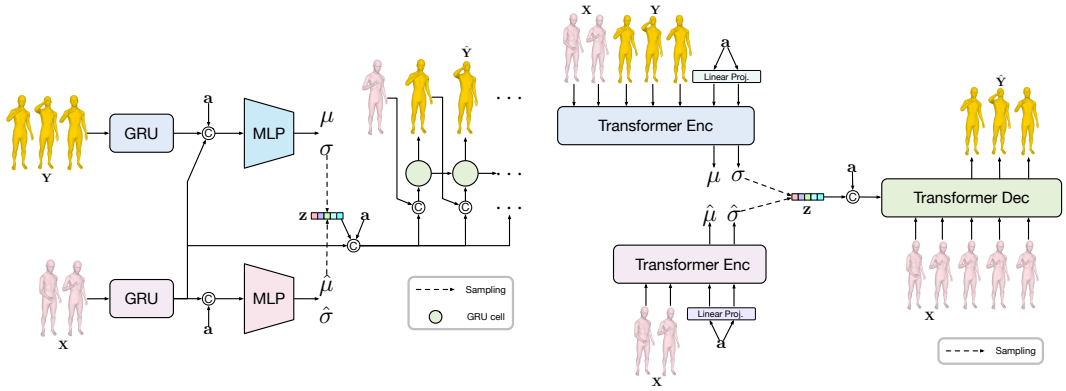
where $p_\psi(\mathbf{z}|\mathbf{X}, \mathbf{a})$ is the prior distribution of the latent variable $\mathbf{z}$, modeled by a neural network with parameters ψ, and $KL(\cdot||\cdot)$ is the KL divergence between two distributions. Training the CVAE then aims to maximize the log probability $\log p(\mathbf{Y}|\mathbf{X}, \mathbf{a})$ by maximizing the ELBO.

In practice, the KL divergence term in the ELBO can be computed as,

$$\begin{aligned} \mathcal{L}_{\text{KL}} &= KL(\mathcal{N}(\mu, \text{diag}(\sigma^2)) \| \mathcal{N}(\hat{\mu}, \text{diag}(\hat{\sigma}^2))) \\ &= \frac{1}{2} \sum_{i=1}^{D} \left(\log \frac{\hat{\sigma}_i{}^2}{\sigma_i^2} + \frac{\sigma_i^2 + (\mu_i - \hat{\mu}_i)^2}{\hat{\sigma}_i^2} - 1 \right) \,, \end{aligned} \quad (2)$$

where $\mathcal{N}(\mu, \text{diag}(\sigma^2))$ and $\mathcal{N}(\hat{\mu}, \text{diag}(\hat{\sigma}^2))$ are the posterior and prior distributions, whose means and standard deviations are produced by the encoder q_ϕ and the prior network p_ψ, respectively, and D is the dimension of $\mathbf{z}$.

During training, the random variable $\mathbf{z}$ is sampled from the posterior distribution via the reparameterization trick [24], i.e., $\mathbf{z} = \epsilon \odot \sigma + \mu$, where $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Given $\mathbf{z}$, the past poses $\mathbf{X}$ and the action label $\mathbf{a}$, the goal of decoder p_θ is to reconstruct the true future motion. This lets



(a) RNN based model (b) Transformer based model

Figure 3. **Network structure.** We explore the use of two different temporal encoding structures to build our VAE: RNNs and Transformers.

us express the (negative of the) first term of the ELBO as the reconstruction loss

$$\mathcal{L}_{\text{rec}} = \frac{1}{T}\sum_{i=1}^{T} \|\hat{\mathbf{y}}_i - \mathbf{y}_i\|_2^2 \,, \tag{3}$$

where $\hat{\mathbf{Y}} = [\hat{\mathbf{y}}_1, \hat{\mathbf{y}}_2, \cdots, \hat{\mathbf{y}}_T]$ is the future motion generated by the decoder.

Note that, during training, sampling $\mathbf{z}$ involves the encoder q_ϕ, which relies on the ground-truth motion $\mathbf{Y}$. Since, at test time, the ground-truth future motion is unknown, we sample the random variable from the prior distribution.

3.2. Weakly-supervised Transitions Learning

Natural human movement involves transitions between different action categories. The ability to generate these transitions is therefore critical for the success and realism of human motion modeling methods. However, acquiring training data that covers all possible action transitions is virtually intractable, and thus existing human motion datasets typically contain motions depicting individual actions only, without any transitions. To nonetheless effectively leverage this data to learn action transitions, we create synthetic motions by combining historical motions from one action category with future motions from another. As these synthetic motions still do not contain realistic transitions, we introduce a weakly-supervised training strategy to learn to generate plausible transitions.

More specifically, given a historical motion $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N]$ from one action, we take motion $\mathbf{Y}' = [\mathbf{y}'_1, \mathbf{y}'_2, \cdots, \mathbf{y}'_T]$ from another action to be the continuation of $\mathbf{X}$ after T_0 frames. However, both the number of frames T_0 and poses in these frames are unknown, and one cannot assume T_0 to be constant for any pair of historical and future motions. To address this, we define T_0 to be a function of the last pose of $\mathbf{X}$ and of the first one of $\mathbf{Y}'$, i.e., $T_0 = f(\mathbf{x}_N, \mathbf{y}'_1)$, where $f : \mathbb{R}^K \times \mathbb{R}^K \to \mathbb{N}$. In practice, we found that a simple linear function suffices, and thus write $T_0 = \lfloor k\|\mathbf{x}_N - \mathbf{y}'_1\|_2 \rfloor$, where $k > 0$ is computed from the training data. Details about computing k are provided in the supplementary material.

To account for the fact that the poses within the transition sequence, namely poses for T_0 frames, are unknown, we leverage a simple temporal smoothness prior based on the intuition that the transition from $\mathbf{X}$ to $\mathbf{Y}'$ should form a smooth sequence. Inspired by [3, 20, 37], we make use of the Discrete Cosine Transform (DCT) to define our smoothness prior, exploiting the insight that a smooth trajectory can be accurately represented by low frequency DCT bases. More precisely, let $\hat{\mathbf{Y}} = [\hat{\mathbf{y}}_1, \hat{\mathbf{y}}_2, \cdots, \hat{\mathbf{y}}_{T_0+T}]$ denote the prediction of our model. We first concatenate the last L poses of the history and the first L ones of the prediction to form a sequence of length $2L$ denoted by $\hat{\mathbf{Z}} = [\mathbf{x}_{N-L+1}, \mathbf{x}_{N-L+2}, \cdots, \mathbf{x}_N, \hat{\mathbf{y}}_1, \hat{\mathbf{y}}_2, \cdots, \hat{\mathbf{y}}_L]$, where $L \leq N$ and $\hat{\mathbf{Z}} \in \mathbb{R}^{K \times 2L}$. We then approximate this sequence with the first M DCT bases as $\tilde{\mathbf{Z}} = \hat{\mathbf{Z}}\mathbf{D}\mathbf{D}^T$, where $\mathbf{D} \in \mathbb{R}^{2L \times M}$ encodes the low-frequency DCT basis and $M \leq 2L$. Given $\hat{\mathbf{Z}}$ and its approximation $\tilde{\mathbf{Z}}$, we define our temporal smoothness prior as the loss

$$\mathcal{L}_{\text{smooth}} = \frac{1}{2L}\sum_{i=1}^{2L} \|\hat{\mathbf{z}}_i - \tilde{\mathbf{z}}_i\|_2^2 \,, \tag{4}$$

where $\hat{\mathbf{z}}_i$ and $\tilde{\mathbf{z}}_i$ are the i-th pose in $\hat{\mathbf{Z}}$ and $\tilde{\mathbf{Z}}$, respectively.

Since we only have ground-truth supervision for the last T predicted frames, we redefine the reconstruction loss as

$$\mathcal{L}_{\text{rec}} = \frac{1}{T}\sum_{i=1}^{T} \|\hat{\mathbf{y}}_{T_0+i} - \mathbf{y}_i\|_2^2 \,. \tag{5}$$

Note that our formulation still allows us to exploit data where $\mathbf{Y}'$ and $\mathbf{X}$ are from the same motion sequence by simply setting the corresponding T_0 to zero.

Altogether, we express our complete training loss as

$$\mathcal{L} = \lambda_{\text{rec}}\mathcal{L}_{\text{rec}} + \lambda_{\text{smooth}}\mathcal{L}_{\text{smooth}} + \mathcal{L}_{\text{KL}} \,, \tag{6}$$

where λ_{rec} and λ_{max} are hyper-parameters setting the relative influence of the different terms.

3.3. Variable-length Motion Prediction

Generating sequences of variable length has been well-studied in the field of Natural Language Processing

(NLP) [50], where the standard strategy consists of predicting a specific stop token. Here, instead of predicting a stop token, which is ill-defined for human motion, we simply encourage the model to generate static poses (the last pose of the ground-truth motion) after reaching the motion end during training. Specifically, we make the model generate P additional frames, leading to a future sequence of T_0+T+P frames ($\hat{\mathbf{Y}} = [\hat{\mathbf{y}}_1, \hat{\mathbf{y}}_2, \cdots, \hat{\mathbf{y}}_{T+T_0+P}]$). We then supervise these additional frames with the last ground-truth future pose. Combining this with the normal supervision of the other frames yields the updated reconstruction loss

$$\mathcal{L}_{\text{rec}} = \frac{1}{T+P} \sum_{i=1}^{T+P} \|\hat{\mathbf{y}}_{T_0+i} - \mathbf{y}_i\|_2^2 \,, \tag{7}$$

During testing, as we do not know the length of the predicted future, we stop prediction when the variance of the last Q consecutive frames falls below a threshold. Specifically, given the predicted future motion $\hat{\mathbf{Y}}$, we compute, for Q consecutive frames starting from the i-th one,

$$v_i = \frac{1}{Q} \sum_{j=i}^{i+Q} \|\hat{\mathbf{y}}_j - \frac{1}{Q} \sum_{k=i}^{i+Q} \hat{\mathbf{y}}_k\|_2 \,, \tag{8}$$

where $i \in [1, 2, \cdots, T_{\text{max}} - Q]$, and T_{max} is the maximum number of frames the model can predict. We stop the prediction at frame i if $v_i < \delta$.

3.4. Network Structure

To show the generality of our approach, we exploit it using two different temporal encoding structures, namely, Recurrent Neural Networks (RNNs) and Transformers [50].

For our RNN-based model, shown in Fig 3 (a), we build the encoder q_ϕ, the prior p_ψ and the decoder p_θ using Gated Recurrent Units (GRUs). In particular, the encoder q_ϕ first uses GRUs to encode the historical human poses $\mathbf{X}$ and the future human motion $\mathbf{Y}$ to temporal features. These temporal features concatenated with the action token obtained from a fully connected layer are then fed into a fully connected network that predicts the parameters (the mean μ and the standard deviations σ) of the posterior distribution. The prior network produces the parameters (the mean $\hat{\mu}$ and the standard deviations $\hat{\sigma}$) of the prior distributions in a similar manner. Given the latent code $\mathbf{z}$ sampled from either the posterior (during training) or the prior (during testing), the temporal features of the historical motion $\mathbf{X}$ and the action label $\mathbf{a}$, the decoder again uses GRUs to predict the future poses in an autoregressive manner.

We show our Transformer-based model in Fig. 3 (b). For the encoder and prior network, we adopt the same strategy as [42], which was inspired by BERT [11] in NLP and ViT [12] in Computer Vision. In particular, we append two extra tokens obtained from the action label $\mathbf{a}$ to aggregate temporal information to predict the parameters of the posterior and prior distributions. For the decoder, we pad the historical human motion with its last pose, forming a longer sequence, and then input the padded sequence to the Transformer-based decoder that outputs the future motion. To introduce the action information and the latent random code as conditions, we further use the pseudo self attention strategy proposed in [58].

4. Experiments

4.1. Datasets

We evaluate our method on three different datasets. Each motion sequence in these datasets is annotated with a single action label, except for BABEL [43]. Some information for each dataset is provided in Table 1. We also evaluate on the dataset HumanAct12 [17]. The results are in the supplementary material.

Dataset	motion len.	train	test	transi.	action
GRAB [7, 47]	100-501	1149	319	0	4
NTU RGBD [33, 44]	35-201	3399	361	0	13
BABEL [43]	30-300	9643	3477	2584	20

Table 1. **Datasets' details.** We list the range of motion length in frames, the number of training/testing samples, the number of training samples with action transitions and the number of actions in each dataset.

GRAB [7, 47] consists of 10 subjects interacting with 51 different objects, performing 29 different actions. Since, for most actions, the number of samples is too small for training, we choose the four action categories with the most motion samples, i.e., Pass, Lift, Inspect and Drink. We use 8 subjects (S1-S6, S9, S10) for training and the remaining 2 subjects (S7, S8) for testing. In all cases, we remove the global translation. The original frame rate is 120 Hz. To further enlarge the size of the dataset, we downsample the sequences to 15-30 Hz. Our model is trained to observe 25 frames to predict the future. The observed frames and the future ones are from either the same or different motions.

NTU RGB-D [33, 44] (NTU). We use the subset of 13 actions of [17], with noisy SMPL parameters estimated by VIBE [25]. As for GRAB, we remove the global translation. While [17] used all the data for training, we split the dataset into training and testing by subjects. Our model is trained to observe 10 past frames.

BABEL [43] is a subset of the AMASS dataset [35] with per-frame action annotations. Since there are multiple action labels in one motion sequence, we split the dataset into two parts: single-action sequences and sequences that depict transitions between two actions. We downsample all motion sequences to 30 Hz. For single-action motions, we first divide the long motions into several short ones. Each short motion performs one single action. and the remove sequences that are too short ($<$ 1 second). We also eliminate the action labels with too few samples ($<$ 60) or overlap with other actions, e.g. foot movement sequences some-

times overlap with kicking. This leaves us with 20 action labels. We complement this data with the sequences with transitions that contain these 20 actions. During training, our model observes 10 past frames to predict the future.

4.2. Evaluation Metrics and Baseline

Metrics. We follow the similar evaluation protocol as for human motion synthesis/prediction [17, 42, 56] and employ the following metrics to evaluate our method.

(1) To measure the distribution similarity between the generated sequences and the ground-truth motions, we adopt the Fréchet Inception Distance (FID) [19]

$$FID = \|\mu_{\text{gen}} - \mu_{\text{gt}}\|^2 + Tr(\Sigma_{\text{gen}} + \Sigma_{\text{gt}} - 2(\Sigma_{\text{gen}}\Sigma_{\text{gt}})^{1/2}) , \quad (9)$$

where $\mu_\cdot \in \mathbb{R}^F$ and $\Sigma_\cdot \in \mathbb{R}^{F\times F}$ are the mean and covariance matrix of perception features obtained from a pretrained action recognition model, with F the dimension of the perception features. The detail of the action recognition model is included in the supplementary material. $Tr(\cdot)$ computes the trace of a matrix.

(2) To evaluate motion realism, we report the action recognition accuracy of the generated motions using the same pretrained action recognition model as above.

(3) To evaluate per-action diversity, we measure the pairwise distance between the multiple future motions generated from the same historical motion and action label[1]. Specifically, given a set of future motions $\{\hat{\mathbf{Y}}^i\}_{i=1}^S$ predicted by our model, the diversity is computed as

$$Div = \frac{2}{S(S-1)} \sum_{i=1}^{S} \sum_{j=i+1}^{S} \frac{1}{T_{\text{max}}} \sum_{k=1}^{T_{\text{max}}} \|\hat{\mathbf{y}}_k^i - \hat{\mathbf{y}}_k^j\|_2 , \quad (10)$$

where T_{max} is the maximum number of frames our model can predict, and $\hat{\mathbf{y}}_k^i$ represents the k-th frame of motion $\hat{\mathbf{Y}}^i$.

To calculate above mentioned diversity, we assume that the model generates the maximum number of future frames in all cases. To further evaluate the diversity across variable length future motions, we compute the average per-action diversity after performing Dynamic Time Warping (DTW) [57] With a minor abuse of notation, let $\{\hat{\mathbf{Y}}^i\}_{i=1}^S$ denote the set of variable length predictions. DTW then temporally aligns any pair of motions as $\tilde{\mathbf{Y}}^i, \tilde{\mathbf{Y}}^j = DTW(\hat{\mathbf{Y}}^i, \hat{\mathbf{Y}}^j)$, where $\tilde{\mathbf{Y}}^i$ and $\tilde{\mathbf{Y}}^j \in \mathbb{R}^{K\times T_{i,j}}$ have the same number of frames ($T_{i,j}$). We then compute the diversity after DTW as

$$Div_{\text{w}} = \frac{2}{S(S-1)} \sum_{i=1}^{S} \sum_{j=i+1}^{S} \frac{1}{T_{i,j}} \sum_{k=1}^{T_{i,j}} \|\tilde{\mathbf{y}}_k^i - \tilde{\mathbf{y}}_k^j\|_2 . \quad (11)$$

(4) To measure the prediction accuracy, we adopt the Average Displacement Error (ADE) computed as

$$ADE = \min_i \frac{1}{T} \sum_{k=1}^{T} \|\hat{\mathbf{y}}_k^i - \mathbf{y}_k\|_2 , \quad (12)$$

where T is the length of the ground-truth future motion, $\hat{\mathbf{y}}_k^i$ is the k-th frame of the i-th sample generated by the model with the ground-truth action label[2] and $\mathbf{y}_k$ the corresponding ground truth. Similarly to the diversity, we report the ADE after DTW (ADE_{w}).

Baselines. Since there is no prior work that tackles the task we introduce, we adapt the state-of-the-art action-specific human motion synthesis methods, Action2Motion [17], ACTOR [42], and stochastic human motion prediction method, DLow [56], to our task. Action2Motion [17] relies on a frame-wise motion VAE with GRUs to encode the temporal information. We adapt their VAE so as to take the temporal feature of historical poses as an additional input for both encoding and decoding. This temporal feature is extracted from a GRU-based temporal data encoding module. Similarly, we modify the transformer decoder of ACTOR [42] to condition it on the historical motion. Furthermore, we adapt the VAE in DLow [56] to take the action label as input.

Implementation details. We implement our models in Pytorch [40] and train them using the ADAM [23] optimizer for 500 epochs. We use different hyperparameters for different models. In particular, for RNN-based model, the initial learning rate is 0.001 on BABEL and 0.002 on all the other datasets. We set the loss weights (λ_{rec}, λ_{smooth}) to (50.0, 10.0) for BABEL dataset and (100.0,100.0) for all the other ones. For Transformer-based model, the initial learning rate is 0.0001 on BABEL and 0.0005 on all the other datasets. The loss weights (λ_{rec}, λ_{smooth}) are set to (100.0, 10.0) for BABEL dataset and (1000.0,100.0) for all the other ones. Additional details are in the supplementary material.

4.3. Results

Quantitative results. In Table 2, we compare our results with those of baselines on GRAB, NTU RGB-D and BABEL. Given a past motion, all models predict multiple future motions conditioned on any given action label. Our approach, based on either RNNs or Transformers, outperforms the baselines on almost all metrics. In general, the RNN-based model performs better than the Transformer-based one. We expect this to be due to the datasets being too small to train the Transformer-based model from scratch.

In Table 3, we compare the prediction accuracy (ADE, ADE_{w}) of our results with the baselines. Here, for each past motion, each model predicts multiple future motions using the ground-truth action label. The prediction accuracy (ADE) is then computed based on the future motion yielding the minimum error. Because our model is trained to predict not only the ground-truth future but also motions with different action labels, it may sacrifice some accuracy when evaluated on the ground-truth future only, as on NTU and BABEL.

During training, our model only takes one action label

[1] Note that we only report diversity per action because motions of different actions are inherently diverse.

[2] Since we only have the ground-truth future motion for the ground-truth action label.

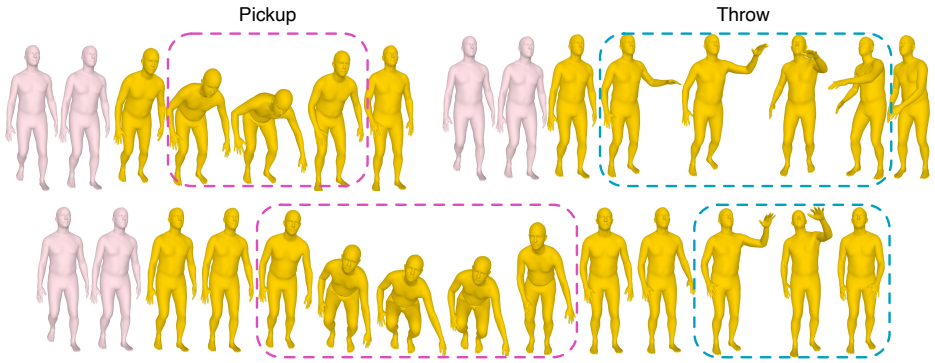


Figure 4. **Results on NTU RGB-D.** Given the same history (pink), our model can generate future motions (yellow) of different actions, e.g., "Pickup" (top left) or "Throw" (top right). Moreover, it can also generate motions depicting a sequence of multiple actions (bottom).

	Method	Acc↑	$FID_{\text{tr}}\downarrow$	$FID_{\text{te}}\downarrow$	$Div_{\text{w}}\uparrow$	$Div\uparrow$
GRAB	Act2Mot [17]	$70.6^{\pm 1.3}$	$80.22^{\pm 6.64}$	$47.81^{\pm 1.09}$	$0.50^{\pm 0.00}$	$0.76^{\pm 0.01}$
	DLow [56]	$67.6^{\pm 0.7}$	$127.49^{\pm 6.90}$	$\mathbf{22.71}^{\pm 2.79}$	$0.74^{\pm 0.01}$	$0.92^{\pm 0.01}$
	ACTOR [42]	$83.0^{\pm 0.3}$	$62.68^{\pm 1.26}$	$114.85^{\pm 3.46}$	$1.06^{\pm 0.00}$	$1.04^{\pm 0.00}$
	Ours (RNN)	$\mathbf{92.6}^{\pm 0.6}$	$\mathbf{44.59}^{\pm 1.39}$	$38.03^{\pm 1.49}$	$\mathbf{1.10}^{\pm 0.01}$	$\mathbf{1.37}^{\pm 0.01}$
	Ours (Tran.)	$85.5^{\pm 1.2}$	$48.58^{3.05}$	$25.72^{\pm 2.16}$	$1.05^{\pm 0.01}$	$1.08^{\pm 0.01}$
NTU	Act2Mot [17]	$66.3^{\pm 0.2}$	$144.98^{\pm 2.44}$	$113.61^{\pm 0.84}$	$0.75^{\pm 0.01}$	$1.19^{\pm 0.01}$
	DLow [56]	$70.6^{\pm 0.2}$	$151.11^{\pm 1.25}$	$157.54^{\pm 1.62}$	$0.97^{\pm 0.00}$	$1.21^{\pm 0.00}$
	ACTOR [42]	$66.3^{\pm 0.1}$	$355.69^{\pm 5.74}$	$193.58^{\pm 2.91}$	$\mathbf{1.84}^{\pm 0.00}$	$2.07^{\pm 0.00}$
	Ours (RNN)	$\mathbf{76.0}^{\pm 0.2}$	$\mathbf{72.18}^{\pm 0.93}$	$\mathbf{111.01}^{\pm 1.28}$	$1.25^{\pm 0.00}$	$\mathbf{2.20}^{\pm 0.00}$
	Ours (Tran.)	$71.3^{\pm 0.2}$	$83.14^{\pm 1.74}$	$114.62^{\pm 0.93}$	$1.25^{\pm 0.00}$	$2.19^{\pm 0.01}$
BABEL	Act2Mot [17]	$14.8^{\pm 0.2}$	$42.02^{\pm 0.40}$	$37.41^{\pm 0.47}$	$0.79^{\pm 0.01}$	$1.10^{\pm 0.01}$
	DLow [56]	$12.7^{\pm 0.2}$	$27.99^{\pm 0.45}$	$24.18^{\pm 0.59}$	$0.65^{\pm 0.00}$	$0.90^{\pm 0.00}$
	ACTOR [42]	$40.9^{\pm 0.2}$	$29.34^{\pm 0.10}$	$30.31^{\pm 0.16}$	$\mathbf{2.94}^{\pm 0.00}$	$\mathbf{2.71}^{\pm 0.00}$
	Ours (RNN)	$\mathbf{49.6}^{\pm 0.4}$	$22.54^{\pm 0.27}$	$22.39^{\pm 0.36}$	$1.35^{\pm 0.00}$	$1.74^{\pm 0.00}$
	Ours (Tran.)	$39.5^{\pm 0.3}$	$\mathbf{20.02}^{\pm 0.24}$	$\mathbf{19.41}^{\pm 0.35}$	$1.39^{\pm 0.00}$	$1.82^{\pm 0.01}$

Table 2. **Quantitative results.** We report the action recognition accuracy (Acc), the FID to training data (FID_{tr}) and to the testing split (FID_{te}), and the diversity before (Div) and after DTW (Div_{w}). We adapt Action2Motion [17], ACTOR [42] and DLow [56] to our task.

and a motion history as input. During testing, to predict future motions for a sequence of action labels of arbitrary length, we follow a recursive strategy. We evaluate this in the case of 5-action sequences. Specifically, we randomly sampled sequences of 5 action labels to generate future motions in an autoregressive manner, and report the results at each prediction step, i.e., corresponding to each action label. The results shown in Table 4 indicate that our model remains stable. Note that the performance gap between the 1st and 2nd step on NTU may be caused by the fact that our model is trained with the jittery "ground-truth" NTU motion history, while at the 2nd step, it starts taking as input the smooth motion predicted during the 1st step.

Qualitative results. In Fig. 1, we show diverse futures generated by our model on GRAB given the same past motion and the same action sequence. Additional qualitative results on the NTU RGB-D dataset are provided in Fig. 4. Given the same historical poses, our model can generate futures of different actions and sequences of multiple actions. More results are provided in the supplementary material.

Trajectory smoothness. We also compare the trajectories produced by different models in Fig. 5 (a). The motions predicted by Action2Motion [17] suffer from heavy

Method	GRAB		NTU		BABEL	
	$ADE_{\text{w}}\downarrow$	$ADE\downarrow$	$ADE_{\text{w}}\downarrow$	$ADE\downarrow$	$ADE_{\text{w}}\downarrow$	$ADE\downarrow$
Act2Mot [17]	$1.92^{\pm 0.03}$	$2.28^{\pm 0.03}$	$\mathbf{0.78}^{\pm 0.01}$	$\mathbf{1.11}^{\pm 0.01}$	$1.25^{\pm 0.02}$	$1.27^{\pm 0.01}$
DLow [56]	$1.78^{\pm 0.03}$	$1.96^{\pm 0.03}$	$0.95^{\pm 0.01}$	$1.20^{\pm 0.01}$	$\mathbf{1.10}^{\pm 0.01}$	$\mathbf{1.19}^{\pm 0.01}$
ACTOR [42]	$2.41^{\pm 0.02}$	$2.57^{\pm 0.02}$	$1.26^{\pm 0.01}$	$1.49^{\pm 0.01}$	$2.19^{\pm 0.02}$	$2.29^{\pm 0.02}$
Ours (RNN)	$1.73^{\pm 0.02}$	$\mathbf{1.93}^{\pm 0.03}$	$0.89^{\pm 0.01}$	$1.20^{\pm 0.01}$	$1.31^{\pm 0.00}$	$1.47^{\pm 0.01}$
Ours (Tran.)	$\mathbf{1.69}^{\pm 0.02}$	$\mathbf{1.93}^{\pm 0.03}$	$0.84^{\pm 0.01}$	$1.23^{\pm 0.01}$	$1.24^{\pm 0.01}$	$1.40^{\pm 0.02}$

Table 3. **Results of prediction accuracy** Our model may trade some performance with GT action label for the ability of predicting future motion of different action labels.

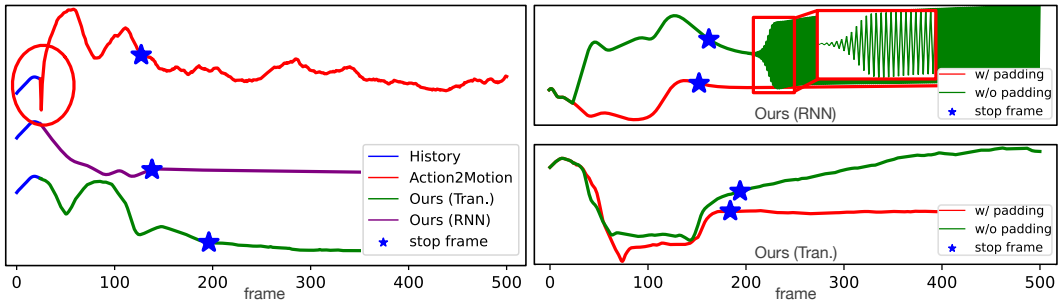
	Metrics	Prediction Step 1_{st}	2_{nd}	3_{rd}	4_{th}	5_{th}
GRAB	Acc↑	$92.6^{\pm 0.6}$	$94.3^{\pm 0.6}$	$93.4^{\pm 0.9}$	$93.5^{\pm 0.6}$	$92.6^{\pm 1.0}$
	$FID_{\text{tr}}\downarrow$	$44.59^{\pm 1.39}$	$31.45^{\pm 6.73}$	$31.53^{\pm 6.36}$	$38.92^{\pm 6.31}$	$43.14^{\pm 10.25}$
	$FID_{\text{te}}\downarrow$	$38.03^{\pm 1.49}$	$74.85^{\pm 13.84}$	$91.65^{\pm 7.21}$	$111.36^{\pm 24.36}$	$117.30^{\pm 13.99}$
	$Div_{\text{w}}\uparrow$	$1.10^{\pm 0.01}$	$1.31^{\pm 0.01}$	$1.33^{\pm 0.01}$	$1.32^{\pm 0.03}$	$1.34^{\pm 0.01}$
	$Div\uparrow$	$1.37^{\pm 0.01}$	$1.60^{\pm 0.01}$	$1.62^{\pm 0.02}$	$1.61^{\pm 0.03}$	$1.64^{\pm 0.02}$
NTU	Acc↑	$76.0^{\pm 0.2}$	$61.9^{\pm 0.7}$	$61.4^{\pm 0.7}$	$60.6^{\pm 0.6}$	$60.1^{\pm 0.6}$
	$FID_{\text{tr}}\downarrow$	$72.18^{\pm 0.93}$	$219.08^{\pm 13.68}$	$248.21^{\pm 13.65}$	$243.57^{\pm 7.11}$	$240.40^{\pm 11.43}$
	$FID_{\text{te}}\downarrow$	$111.01^{\pm 1.28}$	$286.82^{\pm 10.15}$	$334.42^{\pm 16.71}$	$334.94^{\pm 4.53}$	$316.87^{\pm 13.28}$
	$Div_{\text{w}}\uparrow$	$1.25^{\pm 0.00}$	$1.22^{\pm 0.02}$	$1.23^{\pm 0.01}$	$1.22^{\pm 0.02}$	$1.21^{\pm 0.01}$
	$Div\uparrow$	$2.20^{\pm 0.00}$	$2.16^{\pm 0.04}$	$2.18^{\pm 0.01}$	$2.17^{\pm 0.04}$	$2.15^{\pm 0.02}$
BABEL	Acc↑	$49.6^{\pm 0.4}$	$54.4^{\pm 1.0}$	$53.8^{\pm 1.0}$	$55.0^{\pm 1.6}$	$54.4^{\pm 1.7}$
	$FID_{\text{tr}}\downarrow$	$22.54^{\pm 0.27}$	$27.75^{\pm 1.05}$	$27.98^{\pm 0.54}$	$28.10^{\pm 0.52}$	$28.27^{\pm 0.42}$
	$FID_{\text{te}}\downarrow$	$22.39^{\pm 0.36}$	$27.97^{\pm 0.99}$	$28.06^{\pm 0.60}$	$28.32^{\pm 0.65}$	$28.55^{\pm 0.51}$
	$Div_{\text{w}}\uparrow$	$1.35^{\pm 0.00}$	$1.32^{\pm 0.02}$	$1.31^{\pm 0.01}$	$1.29^{\pm 0.01}$	$1.30^{\pm 0.01}$
	$Div\uparrow$	$1.74^{\pm 0.00}$	$1.71^{\pm 0.02}$	$1.69^{\pm 0.01}$	$1.67^{\pm 0.01}$	$1.68^{\pm 0.02}$

Table 4. **Results on prediction with action label sequences.** Our model achieves stable performance at each prediction step.

jitter, especially during the transition between the historical motion and the predicted one (as highlighted by the red circle). The reason is that Action2Motion employs a frame-wise random code, thus making the input to the decoder vary significantly across the frames. Note that this jitter makes our variance-based stopping criterion inapplicable to Action2Motion. We therefore tested different stopping strategies, detailed in the supplementary material, and report the one that gave the best results. When comparing our two models, we found the RNN-based one to produce smoother future motions than the Transformer-based one.

4.4. Ablation Study

To provide a deeper understanding of our model, we evaluate the influence of its two main components, i.e., encouraging the model to predict static poses at the end of the sequence, which we refer to as "padding", and weakly-supervised action transition learning ("weakly-sup"). The results are shown in Table 5. In general, the model with



(a) Action2Motion v.s. Ours (b) Ours w/ padding v.s. w/o padding

Figure 5. **Motion trajectories.** (a) The trajectories from our model are smoother than those of Action2Motion [17]. (b) The trajectories from our model without "padding" either are unstable (top), or do not converge to static poses (bottom).

		Method	Acc↑	FID_{tr} ↓	FID_{te} ↓	Div_w ↑	Div ↑
GRAB	RNN	w/o padding	$88.4^{\pm0.6}$	$\mathbf{32.74}^{\pm0.95}$	$45.61^{\pm1.62}$	$\mathbf{1.14}^{\pm0.01}$	$1.35^{\pm0.01}$
		w/o weakly-sup	$74.2^{\pm0.8}$	$93.84^{\pm1.29}$	$\mathbf{11.42}^{\pm1.26}$	$0.20^{\pm0.00}$	$0.29^{\pm0.00}$
		w/ both	$\mathbf{92.6}^{\pm0.6}$	$44.59^{\pm1.39}$	$38.03^{\pm1.49}$	$1.10^{\pm0.01}$	$\mathbf{1.37}^{\pm0.01}$
	Tran.	w/o padding	$80.4^{\pm0.5}$	$\mathbf{46.38}^{\pm1.45}$	$44.63^{\pm1.19}$	$\mathbf{1.23}^{\pm0.01}$	$\mathbf{1.12}^{\pm0.00}$
		w/o weakly-sup	$48.7^{\pm0.8}$	$184.86^{\pm3.48}$	$\mathbf{23.03}^{\pm1.29}$	$0.01^{\pm0.00}$	$0.01^{\pm0.00}$
		w/ both	$\mathbf{85.5}^{\pm1.2}$	$48.58^{3.05}$	$25.72^{\pm2.16}$	$1.05^{\pm0.01}$	$1.08^{\pm0.01}$
NTU	RNN	w/o padding	$70.1^{\pm0.2}$	$119.25^{\pm0.95}$	$215.69^{\pm2.43}$	$\mathbf{1.34}^{\pm0.00}$	$1.82^{\pm0.01}$
		w/o weakly-sup	$73.6^{\pm0.3}$	$107.88^{\pm2.24}$	$114.00^{\pm0.71}$	$0.53^{\pm0.00}$	$0.89^{\pm0.01}$
		w/ both	$\mathbf{76.0}^{\pm0.2}$	$\mathbf{72.18}^{\pm0.93}$	$\mathbf{111.01}^{\pm1.28}$	$1.25^{\pm0.00}$	$\mathbf{2.20}^{\pm0.00}$
	Tran.	w/o padding	$69.1^{\pm0.1}$	$101.22^{\pm1.65}$	$118.44^{\pm2.07}$	$1.21^{\pm0.00}$	$1.62^{\pm0.00}$
		w/o weakly-sup	$64.7^{\pm0.2}$	$216.56^{\pm2.96}$	$264.92^{\pm6.27}$	$0.02^{\pm0.00}$	$0.02^{\pm0.00}$
		w/ both	$\mathbf{71.3}^{\pm0.2}$	$\mathbf{83.14}^{\pm1.74}$	$\mathbf{114.62}^{\pm0.93}$	$\mathbf{1.25}^{\pm0.00}$	$\mathbf{2.19}^{\pm0.01}$
BABEL	RNN	w/o padding	$46.3^{\pm0.2}$	$39.08^{\pm0.21}$	$37.33^{\pm0.29}$	$\mathbf{1.54}^{\pm0.00}$	$\mathbf{1.78}^{\pm0.00}$
		w/o weakly-sup	$15.6^{\pm0.1}$	$\mathbf{17.67}^{\pm0.41}$	$\mathbf{15.57}^{\pm0.41}$	$0.05^{\pm0.00}$	$0.09^{\pm0.00}$
		w/ both	$\mathbf{49.6}^{\pm0.4}$	$22.54^{\pm0.27}$	$22.39^{\pm0.36}$	$1.35^{\pm0.00}$	$1.74^{\pm0.00}$
	Tran.	w/o padding	$37.8^{\pm0.3}$	$28.70^{\pm0.31}$	$27.64^{\pm0.45}$	$1.38^{\pm0.00}$	$1.61^{\pm0.01}$
		w/o weakly-sup	$12.3^{\pm0.2}$	$20.76^{\pm0.26}$	$\mathbf{17.62}^{\pm0.46}$	$0.01^{\pm0.00}$	$0.01^{\pm0.00}$
		w/ both	$\mathbf{39.5}^{\pm0.3}$	$\mathbf{20.02}^{\pm0.24}$	$19.41^{\pm0.35}$	$\mathbf{1.39}^{\pm0.00}$	$\mathbf{1.82}^{\pm0.01}$

Table 5. **Ablation studies** on generating additional static frames for variable length prediction (padding) and weakly-supervised action transition learning (weakly-sup). Note that, without "weakly-sup", the Transformer-based models suffer from mode collapse, leading to very low motion diversity.

	Method	Acc↑	FID_{tr} ↓	FID_{te} ↓	Div_w ↑	Div ↑
RNN	w/o both	$15.6^{\pm0.1}$	$\mathbf{17.67}^{\pm0.41}$	$\mathbf{15.57}^{\pm0.41}$	$0.05^{\pm0.00}$	$0.09^{\pm0.00}$
	w/ gt-transi	$16.4^{\pm0.4}$	$21.28^{\pm0.31}$	$18.83^{\pm0.34}$	$0.07^{\pm0.00}$	$0.11^{\pm0.00}$
	w/ weakly-sup	$\mathbf{49.6}^{\pm0.4}$	$22.54^{\pm0.27}$	$22.39^{\pm0.36}$	$\mathbf{1.35}^{\pm0.00}$	$\mathbf{1.74}^{\pm0.00}$
	w/ both	$48.4^{\pm0.5}$	$22.70^{\pm0.23}$	$22.47^{\pm0.33}$	$1.31^{\pm0.00}$	$1.71^{\pm0.01}$
Tran.	w/o both	$12.3^{\pm0.2}$	$20.76^{\pm0.26}$	$17.62^{\pm0.46}$	$0.01^{\pm0.00}$	$0.01^{\pm0.00}$
	w/ gt-transi	$13.0^{\pm0.3}$	$20.40^{\pm0.37}$	$\mathbf{17.56}^{\pm0.48}$	$0.01^{\pm0.00}$	$0.01^{\pm0.00}$
	w/ weakly-sup	$\mathbf{39.5}^{\pm0.3}$	$\mathbf{20.02}^{\pm0.24}$	$19.41^{\pm0.35}$	$1.39^{\pm0.00}$	$1.82^{\pm0.01}$
	w/ both	$38.5^{\pm0.3}$	$20.79^{\pm0.27}$	$20.12^{\pm0.39}$	$\mathbf{1.41}^{\pm0.00}$	$\mathbf{1.86}^{\pm0.01}$

Table 6. **Ablation studies** on training with ground truth transition v.s. our weakly-supervised action transition learning.

both components achieves the best performance across all datasets for both temporal encoding structures. Although the numerical results of the models without padding are close to those with padding, we observed the trajectories generated by such models to occasionally either be unstable (RNN-based model) or not converge to a static pose (Transformer-based model), as shown in Fig. 5 (b). Without our weakly-supervised transition learning, the models often fail to produce diverse future motions and the Transformer-based model suffers from mode collapse.

Finally, we compare the performance of using the ground-truth transitions ("gt-transi") to that of our weakly-supervised strategy ("weakly-sup") on BABEL in Table 6. Since the limited ground-truth transitions in BABEL do not cover all possible cases, using them as supervision is ineffective. Specifically, as shown in Table 1, there are only around 2500 ground-truth transition sequences, depicting 170 types of transitions. By contrast, our weakly-supervised strategy leverages almost 100,000 pseudo transitions covering all 380 possible types. This further evidences the importance of our weakly-supervised action transition learning strategy.

5. Conclusion

In this paper, we have introduced the task of *action-driven stochastic human motion prediction*, which aims to predict future trajectories of a given action category. Since it is unrealistic to expect a human motion dataset to include all possible action transitions, we have introduced a weakly-supervised training procedure to learn those transitions from a dataset with only a single action label per sequence. Furthermore, we have introduced a variance-based strategy to produce motions of variable length. Our current model can only generate motions of actions observed in the training set, thus not allowing us to explore novel actions at test time. We will seek to address this in our future work.

Limitations & Negative Societal Impacts

One limitation of our work arises from the fact that our model does not predict global translations. The human movements include local body motions and global translations. However, without scene context, we cannot ensure a valid global translation. For example, a "sit" motion needs to result in sitting on a chair (or something) of the scene.

A potential risk of applying our method to real scenario is that, without considering the scene context, the predicted human motions can lead to unsafe situations, such as collision. We recommend to validate the outputs of our model w.r.t. the environment before applying them to robots/agents.

Acknowledgements

This research was supported in part by the Australia Research Council DECRA Fellowship (DE180100628) and ARC Discovery Grant (DP200102274). The authors would like to thank NVIDIA for the donated GPU (Titan V).

References

[1] Yazan Abu Farha, Alexander Richard, and Juergen Gall. When will you do what?-anticipating temporal occurrences of activities. In *CVPR*, pages 5343–5352, 2018. 3

[2] Chaitanya Ahuja and Louis-Philippe Morency. Language2pose: Natural language grounded pose forecasting. In *3DV*, pages 719–728. IEEE, 2019. 2, 3

[3] Ijaz Akhter, Yaser Sheikh, Sohaib Khan, and Takeo Kanade. Nonrigid structure from motion in trajectory space. In *NeurIPS*, pages 41–48, 2009. 4

[4] Emre Aksan, Manuel Kaufmann, and Otmar Hilliges. Structured prediction helps 3d human motion modelling. In *ICCV*, pages 7144–7153, 2019. 2

[5] Sadegh Aliakbarian, Fatemeh Sadat Saleh, Mathieu Salzmann, Lars Petersson, and Stephen Gould. A stochastic conditioning scheme for diverse human motion prediction. In *CVPR*, pages 5223–5232, 2020. 1, 2

[6] Emad Barsoum, John Kender, and Zicheng Liu. Hp-gan: Probabilistic 3d human motion prediction via gan. In *CVPRW*, pages 1418–1427, 2018. 2

[7] Samarth Brahmbhatt, Cusuh Ham, Charles C. Kemp, and James Hays. ContactDB: Analyzing and predicting grasp contact via thermal imaging. In *CVPR*, 2019. 5

[8] Matthew Brand and Aaron Hertzmann. Style machines. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 183–192. ACM Press/Addison-Wesley Publishing Co., 2000. 1

[9] Judith Butepage, Michael J. Black, Danica Kragic, and Hedvig Kjellstrom. Deep representation learning for human motion prediction and classification. In *CVPR*, July 2017. 2

[10] Yujun Cai, Lin Huang, Yiwei Wang, Tat-Jen Cham, Jianfei Cai, Junsong Yuan, Jun Liu, Xu Yang, Yiheng Zhu, Xiaohui Shen, et al. Learning progressive joint propagation for human motion prediction. In *ECCV*, pages 226–242. Springer, 2020. 2

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *NAACL-HLT*, 2019. 5

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 5

[13] Katerina Fragkiadaki, Sergey Levine, Panna Felsen, and Jitendra Malik. Recurrent network models for human dynamics. In *ICCV*, pages 4346–4354, 2015. 1, 2

[14] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014. 2

[15] Anand Gopalakrishnan, Ankur Mali, Dan Kifer, Lee Giles, and Alexander G Ororbia. A neural temporal model for human motion prediction. In *CVPR*, pages 12116–12125, 2019. 2

[16] Liang-Yan Gui, Yu-Xiong Wang, Xiaodan Liang, and José MF Moura. Adversarial geometry-aware human motion prediction. In *ECCV*, pages 786–803, 2018. 2

[17] Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2motion: Conditioned generation of 3d human motions. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2021–2029, 2020. 1, 2, 3, 5, 6, 7, 8

[18] Alejandro Hernandez, Jurgen Gall, and Francesc Moreno-Noguer. Human motion prediction via spatio-temporal inpainting. In *ICCV*, pages 7134–7143, 2019. 2

[19] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*, 2017. 6

[20] Yinghao Huang, Federica Bogo, Christoph Lassner, Angjoo Kanazawa, Peter V Gehler, Javier Romero, Ijaz Akhter, and Michael J Black. Towards accurate marker-less human shape and pose estimation over time. In *3DV*, pages 421–430. IEEE, 2017. 4

[21] Ashesh Jain, Amir Roshan Zamir, Silvio Savarese, and Ashutosh Saxena. Structural-rnn: Deep learning on spatio-temporal graphs. In *CVPR*, pages 5308–5317, 2016. 1, 2

[22] Seung Wook Kim, Yuhao Zhou, Jonah Philion, Antonio Torralba, and Sanja Fidler. Learning to simulate dynamic environments with gamegan. In *CVPR*, pages 1231–1240, 2020. 3

[23] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *ICLR*, 2015. 6

[24] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 2, 3

[25] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. In *CVPR*, pages 5253–5263, 2020. 5

[26] Hema Swetha Koppula and Ashutosh Saxena. Anticipating human activities for reactive robotic response. In *IROS*, page 2071. Tokyo, 2013. 1

[27] Jogendra Nath Kundu, Maharshi Gor, and R Venkatesh Babu. Bihmp-gan: Bidirectional 3d human motion prediction gan. In *AAAI*, volume 33, pages 8553–8560, 2019. 2

[28] Hsin-Ying Lee, Xiaodong Yang, Ming-Yu Liu, Ting-Chun Wang, Yu-Ding Lu, Ming-Hsuan Yang, and Jan Kautz. Dancing to music. *NeurIPS*, 2019. 2, 3

[29] Chen Li, Zhen Zhang, Wee Sun Lee, and Gim Hee Lee. Convolutional sequence to sequence model for human dynamics. In *CVPR*, pages 5226–5234, 2018. 2

[30] Jiaman Li, Yihang Yin, Hang Chu, Yi Zhou, Tingwu Wang, Sanja Fidler, and Hao Li. Learning to generate diverse dance motions with transformer. *arXiv preprint arXiv:2008.08171*, 2020. 2, 3

[31] Ruilong Li, Shan Yang, David A Ross, and Angjoo Kanazawa. Learn to dance with aist++: Music conditioned 3d dance generation. *arXiv preprint arXiv:2101.08779*, 2021. 2, 3

[32] Xiao Lin and Mohamed R Amer. Human motion modeling using dvgans. *arXiv preprint arXiv:1804.10652*, 2018. 2, 3

[33] Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C Kot. Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. *TPAMI*, 42(10):2684–2701, 2019. 5

[34] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, Oct. 2015. 3

[35] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. Amass: Archive of motion capture as surface shapes. In *ICCV*, Oct 2019. 5

[36] Wei Mao, Miaomiao Liu, and Mathieu Salzmann. History repeats itself: Human motion prediction via motion attention. In *ECCV*, pages 474–489. Springer, 2020. 1, 2

[37] Wei Mao, Miaomiao Liu, Mathieu Salzmann, and Hongdong Li. Learning trajectory dependencies for human motion prediction. In *ICCV*, pages 9489–9497, 2019. 2, 4

[38] Julieta Martinez, Michael J. Black, and Javier Romero. On human motion prediction using recurrent neural networks. In *CVPR*, July 2017. 2

[39] Dirk Ormoneit, Michael J Black, Trevor Hastie, and Hedvig Kjellström. Representing cyclic human motion using functional analysis. *Image and Vision Computing*, 23(14):1264–1276, 2005. 2

[40] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NeurIPS-W*, 2017. 6

[41] Dario Pavllo, David Grangier, and Michael Auli. Quaternet: A quaternion-based recurrent model for human motion. In *BMVC*, 2018. 2

[42] Mathis Petrovich, Michael J. Black, and Gül Varol. Action-conditioned 3D human motion synthesis with transformer VAE. In *ICCV*, pages 10985–10995, October 2021. 1, 2, 3, 5, 6, 7

[43] Abhinanda R. Punnakkal, Arjun Chandrasekaran, Nikos Athanasiou, Alejandra Quiros-Ramirez, and Michael J. Black. BABEL: Bodies, action and behavior with english labels. In *CVPR*, June 2021. 2, 5

[44] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *CVPR*, pages 1010–1019, 2016. 5

[45] Eli Shlizerman, Lucio Dery, Hayden Schoen, and Ira Kemelmacher-Shlizerman. Audio to body dynamics. In *CVPR*, pages 7574–7583, 2018. 2, 3

[46] Sebastian Starke, He Zhang, Taku Komura, and Jun Saito. Neural state machine for character-scene interactions. *ACM Trans. Graph.*, 38(6):209–1, 2019. 1

[47] Omid Taheri, Nima Ghorbani, Michael J. Black, and Dimitrios Tzionas. GRAB: A dataset of whole-body human grasping of objects. In *ECCV*, 2020. 5

[48] Raquel Urtasun, David J Fleet, and Neil D Lawrence. Modeling human locomotion with topologically constrained latent variable models. In *Workshop on Human Motion*, pages 104–118. Springer, 2007. 2

[49] Herwin Van Welbergen, Ben JH Van Basten, Arjan Egges, Zs M Ruttkay, and Mark H Overmars. Real time animation of virtual humans: a trade-off between naturalness and control. In *Computer Graphics Forum*, volume 29, pages 2530–2554. Wiley Online Library, 2010. 1

[50] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008, 2017. 3, 5

[51] Jacob Walker, Kenneth Marino, Abhinav Gupta, and Martial Hebert. The pose knows: Video forecasting by generating pose futures. In *ICCV*, pages 3332–3341, 2017. 2

[52] Borui Wang, Ehsan Adeli, Hsu-kuang Chiu, De-An Huang, and Juan Carlos Niebles. Imitation learning for human pose prediction. In *ICCV*, pages 7124–7133, 2019. 2

[53] Yaohui Wang, Piotr Bilinski, Francois Bremond, and Antitza Dantcheva. Imaginator: Conditional spatio-temporal gan for video generation. In *WACV*, pages 1160–1169, 2020. 3

[54] Xinchen Yan, Akash Rastogi, Ruben Villegas, Kalyan Sunkavalli, Eli Shechtman, Sunil Hadap, Ersin Yumer, and Honglak Lee. Mt-vae: Learning motion transformations to generate multimodal human dynamics. In *ECCV*, pages 265–281, 2018. 2

[55] Ceyuan Yang, Zhe Wang, Xinge Zhu, Chen Huang, Jianping Shi, and Dahua Lin. Pose guided human video generation. In *ECCV*, pages 201–216, 2018. 3

[56] Ye Yuan and Kris Kitani. Dlow: Diversifying latent flows for diverse human motion prediction. In *ECCV*, pages 346–364. Springer, 2020. 1, 2, 6, 7

[57] Jason Y Zhang, Panna Felsen, Angjoo Kanazawa, and Jitendra Malik. Predicting 3d human dynamics from video. In *ICCV*, pages 7114–7123, 2019. 6

[58] Zachary M Ziegler, Luke Melas-Kyriazi, Sebastian Gehrmann, and Alexander M Rush. Encoder-agnostic adaptation for conditional language generation. *arXiv preprint arXiv:1908.06938*, 2019. 5