Can We Use Diffusion Probabilistic Models for 3D Motion Prediction?

Hyemin Ahn[1], Esteve Valls Mascaro[2], Dongheui Lee[2,3]

***Abstract*— After many researchers observed fruitfulness from the recent diffusion probabilistic model, its effectiveness in image generation is actively studied these days. In this paper, our objective is to evaluate the potential of diffusion probabilistic models for 3D human motion-related tasks. To this end, this paper presents a study of employing diffusion probabilistic models to predict future 3D human motion(s) from the previously observed motion. Based on the Human 3.6M and HumanEva-I datasets, our results show that diffusion probabilistic models are competitive for both single (deterministic) and multiple (stochastic) 3D motion prediction tasks, after finishing a single training process. In addition, we find out that diffusion probabilistic models can offer an attractive compromise, since they can strike the right balance between the likelihood and diversity of the predicted future motions. Our code is publicly available on the project website: https://sites.google.com/view/diffusion-motion-prediction.**

I. Introduction

Estimating how a human would move in the near future is an essential task for various applications such as surveillance [1], [2], autonomous driving [3], [4], and human-robot/computer-interaction [5]. Many approaches have been proposed to solve this problem, often based on the motion capture datasets such as Human3.6M [6] or SMPL [7]-based datasets such as AMASS [8]. In this paper, we concern with a task whose goal is to predict a sequence of 3D pose skeletons in Human3.6M and HumanEva-I [9] datasets, when a previously observed 3D pose sequence is given as an input.

Existing works on 3D skeleton motion prediction can be categorized as follows. One line of research focuses on models for deterministic motion prediction [10]–[15]. These works aim at predicting a single motion that is most likely to be observed in the future. Therefore, their performance is usually evaluated based on an $L2$-distance between a prediction and a ground truth. Another line of research focuses on generative models for stochastic motion prediction [16]–[19]. Their performance is evaluated based on the metrics for likelihood and diversity. After generating a fixed number of prediction samples from a single observation, the likelihood is measured based on the minimum distance between the prediction samples and ground truth, and the diversity is measured based on the average distance between all pairs of prediction samples.

[1]Hyemin Ahn is with Artificial Intelligence Graduate School (AIGS), Ulsan National Institute of Science and Technology (UNIST), Ulsan, Korea (e-mail: hyemin.ahn@unist.ac.kr).

[2]Esteve Valls Mascaro and Dongheui Lee are with Autonomous Systems, Technische Universität Wien (TU Wien), Vienna, Austria (e-mail: {esteve.valls.mascaro, dongheui.lee}@tuwien.ac.at).

[3]Dongheui Lee is also with the Institute of Robotics and Mechatronics (DLR), German Aerospace Center, Wessling, Germany.

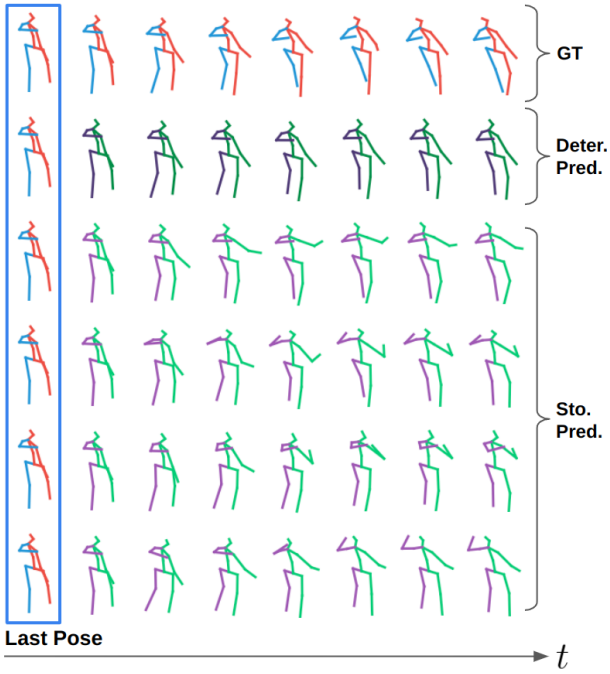

Fig. 1. Example results when diffusion probabilistic models are used for 3D human motion prediction tasks, when observed motion is 'walking'. After a single training procedure, diffusion models can be effectively used for both deterministic (Deter.) and stochastic (Sto.) motion prediction tasks.

However, we cannot judge which approach is always better than the other, since the efficiency would depend on the target application values. For instance, when one needs only the most precise sample with low latency, deterministic approaches would be better. If we say that both approaches are necessary, our next question would be whether we can propose an efficient model for both types of prediction. To answer the question, we study the possibility of using diffusion probabilistic models [20], [21] for both deterministic and stochastic 3D motion prediction tasks.

If we propose a diffusion probabilistic model [20], [21] as a solution, one might ask us whether this is because we are fascinated by its performance in image generation [22], [23]. Frankly speaking, yes, we initiated this study out of our curiosity – can we use diffusion probabilistic models for 3D motion prediction? Unfortunately, our experimental results show that the diffusion model cannot perfectly replace existing state-of-the-arts for both deterministic and stochastic motion prediction tasks. However, we found a glimpse of hope in diffusion models, due to their effectiveness in both prediction types after a single training procedure, and their ability to properly balance the trade-off between diversity and likelihood.

 DOI: 10.1109/ICRA48891.2023.10160722

Figure 1 shows the example results when the diffusion models are used for both deterministic and generative motion prediction tasks. Although a diffusion model is essentially a generative model, we found that the deterministic sample with a fair performance can be obtained from the diffusion model when all randomness is excluded from its denoising process. In addition, we found out that the diffusion models can fix the flaws of several generative methods [18], which highlight the diversity of generated samples. Existing works as [18] claim that the likelihood of predicted samples is high when the minimum distance between samples and ground truth is low. Because of this, [18] can often generate the motions that are out-of-context as [24] pointed out. Compared to this, our diffusion models can generate prediction samples that are more likely to occur, so the generated motion does not diverge too much to be called out-of-context.

The remaining paper is constructed as follows. After representing our literature survey in Section II, Section III will explain how general diffusion models work as well as how we design ours to solve 3D motion prediction tasks. Section IV will show both qualitative and quantitative experiment results, and a related discussion will be also presented. Finally, this paper will end in Section V by mentioning limitations and future works.

II. Related Work

A. 3D Motion Prediction

Deterministic Models. The goal of deterministic 3D motion prediction is to minimize the distance between a predicted motion and ground truth. To solve this problem, early works relying on deep neural networks [10]–[12] often employed recurrent neural networks (RNNs) [25], [26], which are still well-known for their effectiveness in processing time-series data. Among RNN-based works, a notable model is a structure RNN (S-RNN) [12], which considers the spatio-temporal information of human motion, by manually designing the high-level spatio-temporal graph to explicitly model the human body structure (i.e., spine, arm, and leg).

While S-RNN understands the human body structure based on the handcrafted network structure, there is another line of research [13], [27] that uses graph convolutional network (GCN) to overcome this manually designed spatial relationship understanding. For instance, [13] suggested a model named DCT-GCN, where discrete cosine transform (DCT) understands the temporal information of motion, and GCN learns the spatial relationship between human body joints. DCT-GCN obtains the state-of-the-art result when evaluated on Euler-angle-based mean squared error, but its best result can be obtained when the model is separately trained for each short- or long-term prediction.

Recently, several works for deterministic motion prediction [14], [15] are based on the Transformer [28], which was originally suggested for language understanding problems. Models named Spatio-Temporal Transformer (ST-TR) [14] and 2-Channel Transformer (2CH-TR) [15], understand the spatio-temporal relationship of human motion by putting the self-attention mechanism on each pose-parameter (spatial) and time (temporal) dimension. After understanding each spatial and temporal information in parallel, outputs from both attention mechanisms are properly combined. The difference between ST-TR and 2CH-TR comes from when and how often the model combines spatial and temporal information.

Generative Models. The goal of stochastic 3D motion prediction is to build a generative model which can sample out several future motions that are likely to happen after the observed human motion. To solve this problem, early works [16], [17] employed deep generative models such as variational autoencoders (VAEs) [29] or generative adversarial networks (GANs) [30]. For instance, [17] suggested a generative model based on the conditional VAEs, and showed that VAEs can sample out several future motions that are reasonable as well as diverse. Compared to VAE, [16] showed that GANs based on the Wasserstein loss function can be effectively used in stochastic motion prediction tasks.

While these works [16], [17] focused on exploring the potential of using deep generative models in stochastic motion prediction tasks, another line of works [18], [19] focused on sampling out as much as diverse motions that can contain the most plausible motion at the same time. For instance, [18] proposed to train a post-hoc model which can be attached to the pre-trained deep generative model. This post-hoc model maps a random variable to several latent vectors of the pre-trained generative model. Based on the *diversity-promoting prior*, the post-hoc model is trained to improve the diversity between samples, which can be obtained by decoding the mapped latent vectors.

Experiments in [18], [19] evaluate the likelihood of prediction samples based on the *minimum* distance between the samples and the ground truth(s). They denote the prediction samples as plausible based on the sample that is closest to the ground truth(s). However, this can make it difficult for users to choose the most plausible motion among the prediction samples, since all samples will not be distributed near the most plausible motion. For instance, if the observed motion is a human sitting down and drinking something, [18] and [19] can produce motion samples that predict the human suddenly standing up and starting discussing something with others. As [24] has pointed out, we would like to also focus on the necessity of contextually plausible and diverse motion sampling. Therefore, our paper would evaluate the likelihood of prediction also based on the mean and standard deviation of distances between the samples and ground truth.

B. Diffusion Probabilistic Models

Diffusion probabilistic models [20] have become a new rising star in generative models after showing excellent performance in image synthesis. Especially, its performance on text-conditioned image synthesis [22] makes researchers as well as the public in awe. Diffusion models consider two processes: a forward process that slowly destructs the data sample by gradually injecting the random noise, and a reverse process that learns how to reconstruct the data

sample by gradually denoising the random noise. While the advantage of diffusion models can be empirically shown based on their performances, the disadvantage is the speed of their sampling process. If the reverse process includes 1000 times of denoising processes, it means that the data sample can be obtained after feed-forwarding the random noise to the denoising network for 1000 times. Of course, this disadvantage can be circumvented if the application does not require the prediction samples with low latency.

Aside from image generation tasks, nowadays researchers are suggesting to use diffusion models in various generation tasks, such as text-to-speech [31], text-to-sound [32], and video [33]. Focusing on motion-related tasks like ours, several works incorporate diffusion models in text-conditioned motion generation tasks [34], [35]. For the motion of intelligence agents, [36] suggests using diffusion models to sample out trajectories for properly solving a given task. In our paper, we use diffusion models in 3D human motion prediction tasks, but to the best of our knowledge, there is no attempt yet to use diffusion models in the 3D motion prediction task. But we believe more researchers would involve in using diffusion models to answer this question – can diffusion models be our new savior in any kind of data generation tasks?

III. Method

A. Preliminaries

We will provide a short description of diffusion probabilistic models first. Note that our description relies on [20] and [21], which provide a basis for our work.

Diffusion Probabilistic Model. Let $\mathbf{x}^0 \sim q(\mathbf{x}^0)$ denote a data point sampled from its distribution q. In order to learn $p_\theta(\mathbf{x}^0)$ which can model $q(\mathbf{x}^0)$, diffusion probabilistic models consider two processes. One is a *forward process* which gradually deconstructs $\mathbf{x}^0$ by injecting a subtle Gaussian noise for K times, such that $\mathbf{x}^0$ can be destroyed into $\mathbf{x}^1, \dots, \mathbf{x}^K$, where $p(\mathbf{x}^K) = \mathcal{N}(\mathbf{0}, \mathbf{I})$. This process can be formulated as below, which is to follow a Markov chain $q(\mathbf{x}^k|\mathbf{x}^{k-1})$ for K times:

$$q(\mathbf{x}^{1:K}|\mathbf{x}^0) = \prod_{k=1}^{K} q(\mathbf{x}^k|\mathbf{x}^{k-1}) \tag{1}$$

$$q(\mathbf{x}^k|\mathbf{x}^{k-1}) = \mathcal{N}(\sqrt{1-\beta_k}\mathbf{x}^{k-1}, \beta_k\mathbf{I}), \tag{2}$$

where β_k denotes a constant for a noise level. Note that $\mathbf{x}^k$ can be sampled from $\mathbf{x}^0$ directly with a closed-form solution:

$$\mathbf{x}^k = \sqrt{\alpha_k}\mathbf{x}^0 + \sqrt{1-\alpha_k}\boldsymbol{\epsilon}, \;\; \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{3}$$

where $\hat{\alpha}_k = 1 - \beta_k$ and $\alpha_k = \prod_{i=1}^{k} \hat{\alpha}_i$.

Another is a *reverse process*, which goal is to obtain $\mathbf{x}^0$ starting from $\mathbf{x}^K \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, by gradually denoising $\mathbf{x}^K$. This process can also be formulated as following a Markov chain $p_\theta(\mathbf{x}^{k-1}|\mathbf{x}^k)$ for K times:

$$p_\theta(\mathbf{x}^{0:K}) = p(\mathbf{x}^K) \prod_{k=1}^{K} p_\theta(\mathbf{x}^{k-1}|\mathbf{x}^k), \tag{4}$$

$$p_\theta(\mathbf{x}^{k-1}|\mathbf{x}^k) = \mathcal{N}\big(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}^k, k), \sigma^2(k)\mathbf{I}\big), \tag{5}$$

where $p(\mathbf{x}^K) = \mathcal{N}(\mathbf{0}, \mathbf{I})$. To obtain $\boldsymbol{\mu}_\theta$ and σ, [20] suggests denoising diffusion probabilistic models (DDPM), which get $\sigma^2(k) = \frac{1-\alpha_{k-1}}{1-\alpha_k}\beta_k$, parameterize $\boldsymbol{\mu}_\theta$ with θ, and sample $\mathbf{x}^{k-1} \sim p_\theta(\mathbf{x}^{k-1}|\mathbf{x}^k)$ as below:

$$\boldsymbol{\mu}_\theta(\mathbf{x}^k, k) = \frac{1}{\sqrt{\hat{\alpha}_k}}\left(\mathbf{x}^k - \frac{\beta_k}{\sqrt{1-\alpha_k}}\boldsymbol{\epsilon}_\theta(\mathbf{x}^k, k)\right). \tag{6}$$

$$\mathbf{x}^{k-1} = \boldsymbol{\mu}_\theta(\mathbf{x}^k, k) + \sigma(k)\mathbf{z}, \;\; \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{7}$$

In practice, $\boldsymbol{\epsilon}_\theta$ is modeled with a neural network, and it learns how much to denoise from $\mathbf{x}^k$. To train this, [20] suggested a simplified loss function as below:

$$\begin{aligned}\mathcal{L}(\theta) &= \|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{x}^k, k)\|^2 \\ &= \|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\alpha_t}\mathbf{x}^0 + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}, k)\|^2.\end{aligned} \tag{8}$$

In a training process, k is randomly sampled to obtain $\mathcal{L}(\theta)$. For more details, please refer to [20] and [21].

Conditional Diffusion Model. A conditional score-based diffusion model for imputation (CSDI) [21] is proposed to solve a time-series imputation problem using diffusion models. It adds conditional information $\mathbf{x}_{co}$ to eq. (4)-(5):

$$p_\theta(\mathbf{x}^{0:K}) = p(\mathbf{x}^K) \prod_{k=1}^{K} p_\theta(\mathbf{x}^{k-1}|\mathbf{x}^k, \mathbf{x}_{co}), \tag{9}$$

$$p_\theta(\mathbf{x}^{k-1}|\mathbf{x}^k, \mathbf{x}_{co}) = \mathcal{N}(\mathbf{x}^{k-1}; \boldsymbol{\mu}_\theta(\mathbf{x}^k, k|\mathbf{x}_{co}), \sigma^2(k)\mathbf{I}) \tag{10}$$

To define $\boldsymbol{\mu}_\theta(\mathbf{x}^k, k|\mathbf{x}_{co})$, eq. (6)-(7) can be rewritten by adding $\mathbf{x}_{co}$ as a condition to $\boldsymbol{\mu}_\theta$ and $\boldsymbol{\epsilon}_\theta$. Note that $\boldsymbol{\epsilon}_\theta(\mathbf{x}^k, k|\mathbf{x}_{co})$ is modeled with a neural network to learn how much to denoise from $\mathbf{x}^k$ given $\mathbf{x}_{co}$. When training the network, the same loss function as eq. (8) is used, by replacing $\boldsymbol{\epsilon}_\theta$ properly with $\mathbf{x}_{co}$ as a condition.

B. Problem Formulation

Let $\mathbf{p}_t \in \mathbb{R}^D$ be a 3D pose vector at time t, which can be denoted with various representations such as axis-angle, Euler-angle, or xyz-position. Here, $D = 3n$ and n denotes the number of joints. A task of 3D human motion prediction can be defined as predicting future L poses, $P_{pre} = \{\mathbf{p}_{T+1}, \dots \mathbf{p}_{T+L}\} \in \mathbb{R}^{L\times D}$, when T poses, $P_{obs} = \{\mathbf{p}_1, \dots \mathbf{p}_T\} \in \mathbb{R}^{T\times D}$ are observed.

We utilize CSDI [21] for obtaining P_{pre} from given P_{obs}. Starting from $P^0_{pre} = P_{pre}$, our forward process can obtain P^k_{pre} as below:

$$P^k_{pre} = \sqrt{\alpha_k}P^0_{pre} + \sqrt{1-\alpha_k}\boldsymbol{\epsilon}, \;\; \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{11}$$

For a reverse process, we propose a denoiser network which models $\boldsymbol{\epsilon}_\theta(\mathbf{x}^k, k|\mathbf{x}_{co}) = \boldsymbol{\epsilon}_\theta(P^k_{pre}, k|P_{obs})$. This network is trained by minimizing $\mathcal{L}(\theta) = \|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(P^k_{pre}, k|P_{obs})\|^2$.

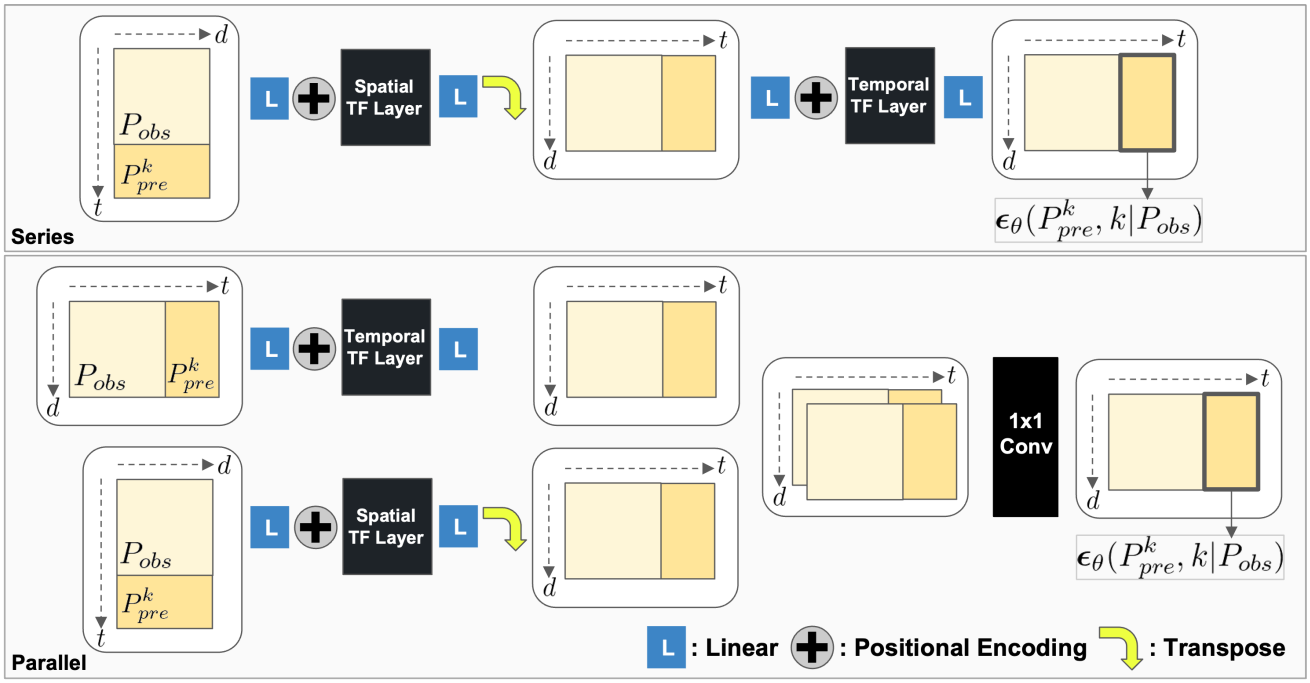

Fig. 2. Two designs of our Transformer-based motion denoiser. Inspired by ST-TR [14], 2CH-TR [15] and CSDI [21], our motion denoiser processes both spatial and temporal information in series (top) or in parallel (bottom). Here, d and t stand for the dimension of each pose-parameter and time, and TF stands for Transformer [28]. Note that the positional encoding also involves adding a learnable vector that represents a diffusion step k as [21] suggests.

After training, we can sample P^0_{pre} by repeating below reverse process for K times, starting from $P^K_{pre} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$:

$$P^{k-1}_{pre} = \boldsymbol{\mu}_\theta(P^k_{pre}, k|P_{obs}) + \sigma(k)\mathbf{z}, \quad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \qquad (12)$$

where $\boldsymbol{\mu}_\theta(P^k_{pre}, k|P_{obs})$ is defined with $\boldsymbol{\epsilon}_\theta(P^k_{pre}, k|P_{obs})$ and properly modified version of eq. (6). After finishing training, if our denoiser network is used for deterministic prediction in the test phase, we set P^K_{pre} and $\mathbf{z}$ as zero-vectors, such that all randomness in eq. (12) can be ignored.

C. Transformer-based Motion Denoiser

Since P^k_{pre} and P_{obs} are time-series of human pose vectors, one can model $\boldsymbol{\epsilon}_\theta(P^k_{pre}, k|P_{obs})$ with neural network architectures which can understand time-series data. For example, network architectures such as RNNs [25], [26] or Transformers [28] can be candidates. We empirically found out that the denoisers based on the Transformers that process both spatial and temporal information are most effective.

Figure 2 shows how we design our Transformer-based denoisers in two ways. Inspired by [21], the first denoiser shown on top of the figure processes both information in series. After concatenating $P^k_{pre} \in \mathbb{R}^{L\times D}$ and $P_{obs} \in \mathbb{R}^{T\times D}$ such that input can be $P^k_{inp} \in \mathbb{R}^{(T+L)\times D}$, P^k_{inp} passes spatial and temporal transformer layers in series, where each layer applies self-attention to time and pose-parameter dimension. Before passing each transformer layer, positional encoding is added to the input as [28] suggests, with respect to pose-parameter $d \in [0, D]$ (spatial) or time $t \in [0, T]$ (temporal) dimension. Also, the additional learnable positional encoding that projects a diffusion step k into a vector space is added to the input as [21] suggests. Let $P^k_{out} \in \mathbb{R}^{(T+L)\times D}$ denote the output which can be obtained after P^k_{inp} passing two layers. Then, the last $L \times D$ parts from P^k_{out} is obtained as $\boldsymbol{\epsilon}_\theta(P^k_{pre}, k|P_{obs})$, which would be used for denoising P^k_{pre}.

The second denoiser shown on the bottom of Figure 2, is inspired by [14] and [15], and works in parallel to understand spatio-temporal information. After P^k_{inp} passes both spatial and temporal transformer layers in parallel, two matrices with the same size as P^k_{inp} are obtained, and concatenated into a 3rd-order tensor whose size is $2 \times (T + L) \times D$. After this tensor passes 2-dimensional convolutional layer with (1×1)-sized kernel, the output $P^k_{out} \in \mathbb{R}^{(T+L)\times D}$ is obtained. From P^k_{out}, $\boldsymbol{\epsilon}_\theta(P^k_{pre}, k|P_{obs})$ is obtained as same as in the first denoiser.

Note that we do not use encoder-decoder based structure, which encode a set of feature vectors from P_{obs} and decode $\boldsymbol{\epsilon}_\theta(P^k_{pre}, k|P_{obs})$ from the encoded feature vectors and P^k_{pre}. We tried various denoisers of Transformer- or RNN-based encoder-decoder, but none of them turns out to be effective.

D. Implementation Details

Our transformer-based motion denoisers have a self-attention module with 8 multi-heads and 512-dimensional query, key, and value vectors. And each temporal or spatial transformer layer shown in Figure 2 consists of a single-layered transformer encoder. To train denoisers, we set batch size as 512 and update parameters for 50,000 iterations with Adam optimizer of learning rate 0.0001. The diffusion step is set as $k \in [0, 20]$, with linearly scheduled noise levels β_k that ranges between 0.001 ($k \downarrow$) and 0.333 ($k \uparrow$).

IV. Experiment

A. Dataset and Metric

Dataset. We conduct our experiment for both deterministic and stochastic motion prediction tasks. For deterministic experiments, we use the Human3.6M dataset [6] and measure the Euler-angle mean square error (MSE) for evaluation as other works [12]–[15] do. Here, with 25 fps, input observation has 50 frames, and output prediction has 25 frames. For stochastic experiments, we preprocess Human3.6M [6] and HumanEva-I [9] datasets into xyz-based representation as [18], [19] do. Based on that, various metrics for evaluating likelihood and diversity are measured. Here, with 50 fps, an input observation has 25 frames, output prediction has 100 frames, and the number of prediction samples is 50.

Metrics. As mentioned above, we measure the performance of our denoiser based on the Euler-angle MSE when it is used for deterministic prediction. For stochastic prediction, we use several metrics from what [18] suggests to evaluate likelihood and diversity. But we propose more metrics such as aDE, sDE, aFDE, and sFDE to measure how the samples are distributed near the ground truth. Note that some of the below sentences describing metrics are borrowed from [18].

(1) **Average Pairwise Distance (APD)**: average $L2$ distance between pairs from N predictions $\hat{\mathbf{x}} \in \mathbb{R}^{L\times D}$, which is computed as $\frac{1}{N(N-1)}\sum_{i=1}^{N}\sum_{j\neq i}^{N}\|\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j\|_2$. This measures the diversity within N predictions. (2) **minimum Displacement Error (mDE)**: the minimum $L2$ distance between all N predictions $\hat{\mathbf{x}}$ and ground truth $\mathbf{x}$, which is computed as $\min_{\hat{\mathbf{x}}}\frac{1}{L}\|\hat{\mathbf{x}} - \mathbf{x}\|_2$. This metric was defined as ADE in [18]. (3) **average Displacement Error (aDE)**: the average $L2$ distance between all N predictions $\hat{\mathbf{x}}$ and ground truth $\mathbf{x}$, which is computed as $\frac{1}{NL}\sum_{i=1}^{N}\|\hat{\mathbf{x}}_i - \mathbf{x}\|_2$. (4) **standard deviation of Displacement Error (sDE)**: the standard deviation of $L2$ distances between all N predictions and ground truth. (5) **minimum Final Displacement Error (mFDE)**: the minimum $L2$ distance between final poses of N predictions and ground truth, which is calculated as $\min_{\hat{\mathbf{x}}}\|\hat{\mathbf{x}}(L) - \mathbf{x}(L)\|_2$. This metric was defined as FDE in [18]. (6) **average Final Displacement Error (aFDE)**: the average $L2$ distance between final poses of N predictions and ground truth, which is calculated as $\frac{1}{N}\sum_{i=1}^{N}\|\hat{\mathbf{x}}_i(L) - \mathbf{x}(L)\|_2$. (7) **standard deviation of Final Displacement Error (sFDE)**: the standard deviation of $L2$ distances between final poses of N predictions and ground truth.

B. Quantitative Results

Deterministic Prediction. Table I compares Euler-angle MSEs when our diffusion model is used for deterministic motion prediction. Here, bold fonts denote the best results among all approaches, and underlines denote the best results among our denoisers (series or parallel). It is shown that the overall performance of DCT-GCN [13] is still the best. Among our approaches, the denoiser which understands spatial and temporal information in series is better than the parallel denoiser. Although our models do not achieve state-of-the-art results, it is shown that our approaches are better in long-term prediction (1000ms) when compared with other transformer-based models [14], [15]. This is a notable result, since (1) our models are originally generative ones, and (2) our models do not require additional training for deterministic prediction since ignoring all randomness in the denoising process is all they need.

TABLE I
AVERAGE MSE ERRORS OF DETERMINISTIC MOTION PREDICTION

milisecond (ms)	80	160	320	400	560	1000
S-RNN [12]	0.933	1.166	1.397	1.526	1.711	2.139
DCT-GCN [13]	0.295	**0.542**	**0.857**	**0.974**	**1.154**	**1.590**
ST-TR [14]	0.303	0.550	0.901	1.021	1.229	1.722
2CH-TR [15]	**0.293**	0.555	0.893	1.016	1.245	1.744
Ours (Series)	<u>0.325</u>	<u>0.615</u>	<u>0.990</u>	<u>1.128</u>	<u>1.309</u>	1.721
Ours (Parallel)	0.350	0.646	1.007	1.148	1.317	<u>1.688</u>

Stochastic Prediction Table II shows the comparison of metrics for measuring the likelihood and diversity. Here, bold fonts denote the best result and underlines denote the second best result among all approaches. It is shown that previous works [18], [19] focusing on sample diversity best perform in APD. Also, it is shown that they are generally better in terms of mDE and mFDE. We would like to argue here that the high diversity in prediction increases the probability of having one sample closest to the ground truth. Then, how can we choose the most plausible result among predictions that are sampled to be diverse?

This is the same question that [24] also pointed out. So in [24], metrics for measuring the quality and context are proposed. For measuring the quality, [24] used a pre-trained binary classifier which can discriminate the ground truths (real) from predictions (fake). If this classifier fails to discriminate the predicted motions as fake, a higher quality score is obtained. For measuring the context, [24] used a pre-trained model which classifies action from motion. If it estimates that the action label of prediction is as same as the observed motion, a higher context score is obtained.

However, we were not able to use the same metric as [24] since its pre-trained classifiers were not openly released. Therefore, we instead propose metrics such as aDE, sDE, aFDE, and sFDE, to measure how closely the samples are distributed near the ground truth. Results show that our approaches generally perform better in terms of these new metrics, and the parallel denoiser performs better than the series one. We also present the result from VAEs [29] that were implemented by [18], to check how other non-diffusion generative models work. It is shown that the overall performances of our series/parallel denoiser in diversity and likelihood are generally better than the VAEs, especially in the HumanEva-I dataset.

C. Qualitative Results

Figure 3 shows two example results from our transformer-based motion denoiser. Predictions on the left of the dotted line are obtained from the motion observation labeled as 'smoking'. It is shown that the deterministic prediction is similar to the ground truth, while the stochastic predictions show the diversity between samples. But note that still

TABLE II

DIVERSITY AND LIKELIHOOD METRICS OF STOCHASTIC MOTION PREDICTION

	Human 3.6M [6]							HumanEva-I [9]						
metrics	APD↑	mDE↓	aDE↓	sDE↓	mFDE↓	aFDE↓	sFDE↓	APD↑	mDE↓	aDE↓	sDE↓	mFDE↓	aFDE↓	sFDE↓
DLow [18]	11.741	0.425	0.968	0.355	0.518	1.387	0.541	4.855	0.251	0.585	0.208	0.268	0.710	0.255
VAEs [18], [29]	6.852	0.460	0.720	0.139	0.557	1.025	0.243	2.299	0.265	0.426	0.083	0.299	0.562	0.137
GSPS [19]	**14.757**	**0.389**	1.206	0.623	**0.496**	1.554	0.729	**5.825**	**0.233**	0.655	0.206	0.244	0.763	0.268
Ours (Series)	7.587	0.527	0.764	**0.132**	0.669	1.093	**0.228**	2.746	0.257	0.383	0.065	0.260	0.490	0.130
Ours (Parallel)	6.445	0.477	**0.719**	0.139	0.584	**1.018**	0.234	1.508	0.242	**0.312**	**0.037**	**0.238**	**0.385**	**0.078**

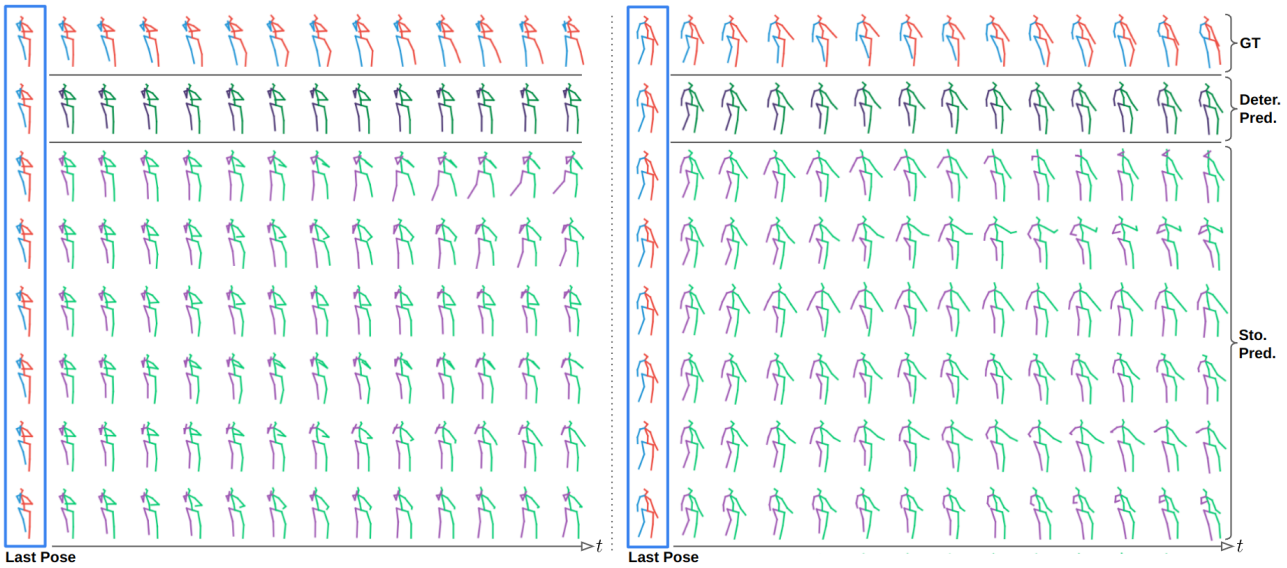

Fig. 3. Deterministic (Deter.) and stochastic (Sto.) predictions from our transformer-based motion denoiser. Note that two results are given and divided based on the vertical dotted line. Predictions are obtained from observed motions labeled as 'smoking' (left) and 'walking' (right).

the context of 'smoking' looks remained in all samples. This phenomenon is also observed from the predictions on the right, which are obtained from the motion observation of 'walking'. While its deterministic prediction resembles the ground truth, the stochastic predictions are diverse and contain the context of 'walking'. For better visualization, please refer to our supplementary video.

V. CONCLUSION

In this work, we study the potential of diffusion probabilistic models for 3D human motion prediction tasks. We propose two types of diffusion models based on the transformers, which understand the motion's spatial and temporal information in series or parallel. Since the diffusion model is originally a generative model, its main usage would be for the stochastic motion prediction task. But once it is trained, we show that it can also be used in deterministic prediction if all randomness in its denoising process is ignored.

To show the effectiveness of diffusion models in both deterministic and stochastic motion prediction tasks, we conduct experiments based on various metrics. Results from deterministic prediction show that the diffusion model is not superior to the state-of-the-art. But it is shown that our long-term (1000ms) prediction performance is better than other transformer-based approaches. When it comes to evaluating stochastic predictions, it is conventional to suggest metrics measuring both likelihood and diversity. However, we claim that the conventional metrics for measuring the likelihood do not represent how much the samples are distributed near the plausible motion, since they measure the *minimum* distance between samples and ground truth. Therefore, we suggest additional metrics to measure the mean and standard deviation of that distances, and the results show that our diffusion models can properly balance the trade-off between diversity and likelihood.

Although our results would provide nice answers to our first question – can we use diffusion probabilistic models for 3D motion prediction? – the most concerning disadvantage of a diffusion model is its sampling frequency. Since our diffusion model requires a $K = 20$ number of denoising processes to obtain prediction samples, this might occur a bit high latency. To overcome this issue, one might consider recent works for efficient sampling [37], which would be our future work, such that efficient 3D human motion prediction can be made for various real-time applications.

ACKNOWLEDGMENT

This work was supported by the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2020-0-01336, Artificial Intelligence Graduate School Program (UNIST)), and funded by Marie Sklodowska-Curie Action Horizon 2020 (Grant agreement No. 955778) for project 'Personalized Robotics as Service Oriented Applications' (PERSEO).

REFERENCES

[1] B. Zhou, X. Tang, and X. Wang, "Learning collective crowd behaviors with dynamic pedestrian-agents," *International Journal of Computer Vision (IJCV)*, vol. 111, no. 1, pp. 50–68, 2015.

[2] Y. Xu, Z. Piao, and S. Gao, "Encoding crowd interaction with deep neural network for pedestrian trajectory prediction," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 5275–5284.

[3] A. Rasouli, I. Kotseruba, T. Kunic, and J. K. Tsotsos, "Pie: A large-scale dataset and models for pedestrian intention estimation and trajectory prediction," in *International Conference on Computer Vision (ICCV)*, 2019, pp. 6262–6271.

[4] K. Kim, Y. K. Lee, H. Ahn, S. Hahn, and S. Oh, "Pedestrian intention prediction for autonomous driving using a multiple stakeholder perspective model," in *International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 7957–7962.

[5] J. Bütepage, H. Kjellström, and D. Kragic, "Anticipating many futures: Online human motion prediction and generation for human-robot interaction," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 4563–4570.

[6] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments," *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 36, no. 7, pp. 1325–1339, jul 2014.

[7] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "Smpl: A skinned multi-person linear model," *ACM transactions on graphics (TOG)*, vol. 34, no. 6, pp. 1–16, 2015.

[8] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black, "Amass: Archive of motion capture as surface shapes," in *International Conference on Computer Vision (ICCV)*, 2019, pp. 5442–5451.

[9] L. Sigal and M. J. Black, "Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion," *Brown Univertsity TR*, vol. 120, no. 2, 2006.

[10] J. Martinez, M. J. Black, and J. Romero, "On human motion prediction using recurrent neural networks," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2891–2900.

[11] K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik, "Recurrent network models for human dynamics," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4346–4354.

[12] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-rnn: Deep learning on spatio-temporal graphs," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 5308–5317.

[13] W. Mao, M. Liu, M. Salzmann, and H. Li, "Learning trajectory dependencies for human motion prediction," in *International Conference on Computer Vision (ICCV)*, 2019, pp. 9489–9497.

[14] E. Aksan, M. Kaufmann, P. Cao, and O. Hilliges, "A spatio-temporal transformer for 3d human motion prediction," in *International Conference on 3D Vision (3DV)*. IEEE, 2021, pp. 565–574.

[15] E. Valls Mascaro, S. Ma, H. Ahn, and D. Lee, "Robust human motion forcasting using transformer-based model," in *International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022.

[16] E. Barsoum, J. Kender, and Z. Liu, "Hp-gan: Probabilistic 3d human motion prediction via gan," in *Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2018, pp. 1418–1427.

[17] X. Yan, A. Rastogi, R. Villegas, K. Sunkavalli, E. Shechtman, S. Hadap, E. Yumer, and H. Lee, "Mt-vae: Learning motion transformations to generate multimodal human dynamics," in *European conference on computer vision (ECCV)*, 2018, pp. 265–281.

[18] Y. Yuan and K. Kitani, "Dlow: Diversifying latent flows for diverse human motion prediction," in *European Conference on Computer Vision (ECCV)*. Springer, 2020, pp. 346–364.

[19] W. Mao, M. Liu, and M. Salzmann, "Generating smooth pose sequences for diverse human motion prediction," in *International Conference on Computer Vision (ICCV)*, 2021, pp. 13 309–13 318.

[20] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 6840–6851, 2020.

[21] Y. Tashiro, J. Song, Y. Song, and S. Ermon, "Csdi: Conditional score-based diffusion models for probabilistic time series imputation," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 34, pp. 24 804–24 816, 2021.

[22] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. Denton, S. K. S. Ghasemipour, B. K. Ayan, S. S. Mahdavi, R. G. Lopes, *et al.*, "Photorealistic text-to-image diffusion models with deep language understanding," *arXiv preprint arXiv:2205.11487*, 2022.

[23] C. Saharia, W. Chan, H. Chang, C. Lee, J. Ho, T. Salimans, D. Fleet, and M. Norouzi, "Palette: Image-to-image diffusion models," in *SIGGRAPH Conference Proceedings*, 2022, pp. 1–10.

[24] S. Aliakbarian, F. Saleh, L. Petersson, S. Gould, and M. Salzmann, "Contextually plausible and diverse 3d human motion prediction," in *International Conference on Computer Vision (ICCV)*, 2021, pp. 11 333–11 342.

[25] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[26] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[27] W. Mao, M. Liu, and M. Salzmann, "History repeats itself: Human motion prediction via motion attention," in *European Conference on Computer Vision (ECCV)*. Springer, 2020, pp. 474–489.

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems (NeurIPS)*, vol. 30, 2017.

[29] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[30] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[31] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, and M. Kudinov, "Grad-tts: A diffusion probabilistic model for text-to-speech," in *International Conference on Machine Learning*. PMLR, 2021, pp. 8599–8608.

[32] D. Yang, J. Yu, H. Wang, W. Wang, C. Weng, Y. Zou, and D. Yu, "Diffsound: Discrete diffusion model for text-to-sound generation," *arXiv preprint arXiv:2207.09983*, 2022.

[33] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, "Video diffusion models," *arXiv preprint arXiv:2204.03458*, 2022.

[34] J. Kim, J. Kim, and S. Choi, "Flame: Free-form language-based motion synthesis & editing," *arXiv preprint arXiv:2209.00349*, 2022.

[35] M. Zhang, Z. Cai, L. Pan, F. Hong, X. Guo, L. Yang, and Z. Liu, "Motiondiffuse: Text-driven human motion generation with diffusion model," *arXiv preprint arXiv:2208.15001*, 2022.

[36] M. Janner, Y. Du, J. B. Tenenbaum, and S. Levine, "Planning with diffusion for flexible behavior synthesis," *arXiv preprint arXiv:2205.09991*, 2022.

[37] T. Salimans and J. Ho, "Progressive distillation for fast sampling of diffusion models," in *International Conference on Learning Representations*, 2021.