Motron: Multimodal Probabilistic Human Motion Forecasting

Tim Salzmann[1], Marco Pavone[2,3] and Markus Ryll[1]
[1] Technical University of Munich [2] Stanford University [3] NVIDIA Research
{tim.salzmann, markus.ryll}@tum.de pavone@stanford.edu

Abstract

Autonomous systems and humans are increasingly sharing the same space. Robots work side by side or even hand in hand with humans to balance each other's limitations. Such cooperative interactions are ever more sophisticated. Thus, the ability to reason not just about a human's center of gravity position, but also its granular motion is an important prerequisite for human-robot interaction. Though, many algorithms ignore the multimodal nature of humans or neglect uncertainty in their motion forecasts. We present Motron, a multimodal, probabilistic, graph-structured model, that captures human's multimodality using probabilistic methods while being able to output deterministic maximum-likelihood motions and corresponding confidence values for each mode. Our model aims to be tightly integrated with the robotic planning-control-interaction loop; outputting physically feasible human motions and being computationally efficient. We demonstrate the performance of our model on several challenging real-world motion forecasting datasets, outperforming a wide array of generative/variational methods while providing state-of-the-art single-output motions if required. Both using significantly less computational power than state-of-the art algorithms.

1. Introduction

The key desideratum of autonomous systems is to provide added-value to humans while ensuring safety. Traditionally, safety aspects limit such robots to low-risk tasks with minimal human interaction. An understanding of humans and their distribution of feasible anticipated movement is key to develop safe, risk-aware human-interactive autonomous systems. Such systems could operate in closer proximity to humans, performing tasks involving higher levels of interaction, providing enhanced added value.

However, capturing the complexity of human motion in a computational model is challenging due to the multitude of continuous movement possibilities (multimodality), even within fixed boundaries of physical limitations. Traditionally, over-conservative systems rely solely on those constraints to ensure safety, while predictive single-motion-output methods discard the potential of many high-level futures in favor of a single possible motion.

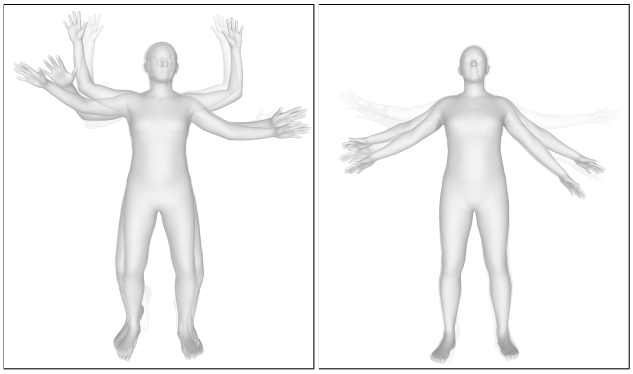


Figure 1. Two examples of multiple possible poses at a given prediction time weighted by their probability. Left: Start of high variance jumping motion. Right: Landing from jump. Lower body is predicted with low uncertainty as feet will come to a stop during landing. Our parametric output distribution captures the full multimodal uncertainty in human motion; enabling subsequent evaluation of samples, individual modes, or most-likely motion as well as their respective confidence values.

In contrast to such deterministic regressors, Monte-Carlo method models aim to produce samples of human future motion. However, for different reasons (e.g., simulation purposes), they prioritize generative diversity over representing actual plausible motions. For robotic use cases, we aim for samples to represent the underlying distribution of possible motions; or even better a parametric description thereof (Fig. 1).

Both, single-output and sampled, human motion predictions have been developed independently to maximize their strengths in accuracy and diversity. Still, many of them have been developed without directly accounting for real-world robotic use cases; they settle for diversity instead of accuracy, ignore physical boundaries, and are computationally too expensive. We target robotic use cases, where predictions are used in a control loop. As such, a model has to capture the underlying distribution of possible human motions without being over-confident or over-diverse within an imminent time horizon; *combining* and *balancing* the desiderata of both research strands. In consequence, we are interested in developing a human motion prediction model that (I) represents the inherent multimodal structure of human motion; (II) accurately captures humans' probabilistic

and diverse nature while obeying rigid skeleton constraints; (III) by directly incorporating structural information of the human skeleton; (IV) while being able to deal with imperfect data; (V) outputting the maximum amount of information for the use of subsequent systems.

To achieve those desiderata, our contribution is threefold: First, we describe a new efficient way to model graph-structured problems where nodes have a fixed semantic class, usable for generic graph-structured problems. Secondly, we present *Motron*, which uniquely uses a probabilistic output structure based on the Concentrated Gaussian distribution in SO3 and a parallel weight sharing approach incorporating the skeleton's structure. It is designed to mirror the multimodal and uncertain nature of humans. *Motron*'s flexible output structure is designed to serve downstream robotic modules such as motion planning, decision making, and control. Finally, we evaluate our model on the fulfillment of our desiderata using single-output metrics *and* metrics based on samples; combining both research strands. We outperform an extensive selection of Monte-Carlo based motion prediction methods while showing state-of-the-art performance on single-output evaluation procedures using a variety of metrics and datasets. Our contributions are substantiated by a thorough ablation study. We further show that *Motron* can deal with occluded data often present in real-world applications by including the additional uncertainty in its output.

2. Related Work

Human Motion Forecasting. The release of the H3.6M dataset [20] in 2014 stimulated a wealth of research in the field of human motion prediction. While early works were based on traditional approaches such as Markov Models [32, 41] or Gaussian Processes [49] the advances of deep learning has taken over recent approaches. Algorithms now are largely based on single-motion-output regressors [11, 12, 38, 40]; improving performance by incorporating human skeleton structural information into their architectures [23, 33]. Li *et al*. [33] merge nodes to create meta graphs of different scale to extract higher level information. Two concepts of capturing temporal influences have emerged: Recurrent Neural Networks (RNNs) and transforming time-series to a frequency domain [1, 35, 54], where the latter is not agnostic to varying history horizons. Recently, Transformer [46] architectures have been explored overcoming RNNs shortcomings in capturing longer time series [35, 45]. While generative and variational approaches have emerged as state-of-the-art in trajectory forecasting [27, 43, 44] for their plethora of captured information, they largely exist in parallel to research on single-output regressors in human motion forecasting. The field is split in approaches using Generative Adversarial Networks (GANs) [4, 17, 18, 28] and (Conditional) Variational Autoencoders ((C)VAEs) [52–54]. Of these, similar to the field of trajectory prediction, adapted versions of the VAE frameworks show better results [53].

Surprisingly, these two fields of single-output and probabilistic motion prediction are highly disentangled, following their respective distinct experimentation protocols. In single-motion-output setups, it is, for example, common to predict one future second. In contrast, previous probabilistic works set their focus on producing plausible diverse motion samples over a longer prediction horizon (2s); at which point the true human motion is fraught with uncertainty.

By slight abuse of terminology, we will distinguish the two common types of motion prediction algorithms in the remainder of this work. Monte-Carlo based models which can produce samples from a distribution of future motions will be referenced as *probabilistic* models while single-motion-output models will be referenced as *deterministic* models for their lack of uncertainty awareness.

Directional Probabilistic Learning. In robotic applications, such as filtering, the use of directional statistics for rotational systems has been proven useful [13, 15, 29, 30]. The most common distributions for modeling rotational uncertainty are the Bingham [6], the von Mises–Fisher [10], the Projected Gaussian [31], and using a Concentrated Gaussian in SO(3)[3, Chapter 7.3.1]. Advances in deep probabilistic learning, however, are mainly focused on learning distributions in vector space [7, 8, 43]. Thus, of the rotational distributions, only the Bingham distribution has been applied to probabilistic deep learning [14, 42]. In [14], Gilitschenski *et al*. directly learn the parameters of a Bingham distribution representing the orientation of an object in an image.

Graph Neural Networks. Besides other concepts like message passing, convolution, and aggregation [55] the concept of attention, first introduced for temporal dependencies [46], has been applied to graph-structured problems by Veličković *et al*. [47] as Graph Attention Networks (GAT). Recently those GATs have been included in temporal networks such as LSTMs by replacing the linear transformations in each RNN cell with a Graph Attention Layer [50]. Within human motion forecasting Graph Convolution Networks (GCNs) [35, 36] have shown good performance. Salzmann *et al*. [43] model dependencies between different entities for trajectory prediction using a sequential message passing algorithm. This, however, becomes computational infeasible for larger number of node types.

3. Problem Formulation

We aim to generate plausible motion distributions for a fixed number N of human skeleton nodes (joints) $n_1, ..., n_N$. Each node n_i is assigned to a semantic class S_i, e.g. Elbow, Knee, or Hip. At time t, given the D-dimensional state $\mathbf{s} \in \mathbb{R}^D$ of each node and all of their histories for the previous H timesteps, which we denote as $\mathbf{x} = \mathbf{s}_{1,...,N}^{(t-H:t)} \in \mathbb{R}^{(H+1)\times N\times D}$, we seek a distribution over all nodes' future states for the next T timesteps $\mathbf{y} = \mathbf{s}_{1,...,N}^{(t+1:t+T)} \in \mathbb{R}^{T\times N\times D}$, which we denote as $p(\mathbf{y} \mid \mathbf{x})$.

4. Preliminaries

Quaternion Representation. The rotational nature of the human anatomy presents a challenge to neural networks. Commonly used rotation representations in $\mathbb{R}^3$, Euler angles and exponential maps, suffer from singularities, discontinuity, and non-uniqueness [16, 40]; all properties contrary to neural network characteristics. Outside of deep learning, however, quaternions have long been established as the default rotational representation. As a consequence of their properties, interpretability, and common use in robotics we choose quaternions as the data representation throughout our model. Thus the input state $\mathbf{x}_i^{(t)} = [\mathbf{q}_i^{(t)}, \dot{\mathbf{q}}_i^{(t)}]$ is defined as the concatenation of the rotation in quaternion representation and its time differential, where $\mathbf{q}_i^{(t)} = \mathbf{q}_i^{(t-1)} \odot \dot{\mathbf{q}}_i^{(t)}$; resulting in a $D = 8$ dimensional state.

Probabilistic Rotations. We use the Concentrated Gaussian distribution $\mathcal{N}_{SO(3)}$ [3, Chapter 7.3.1] to model a probability distribution over the rotation group SO(3). A probabilistic rotation is given as

$$\mathbf{R} = \exp(\epsilon\hat{})\bar{\mathbf{R}} \tag{1}$$

where $\bar{\mathbf{R}}$ is a 'large', noise-free, nominal rotation, exp is the exponential map, $\hat{}$ is a linear, skew-symmetric Lie algebra operator, and $\epsilon \in \mathbb{R}^3$ is a 'small', noisy component:

$$\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}) \tag{2}$$

The distribution's *p.d.f* is defined as

$$p(\mathbf{R} \mid \bar{\mathbf{R}}, \mathbf{\Sigma}) = \frac{1}{Z} e^{-\frac{1}{2}(\ln(\mathbf{R}\bar{\mathbf{R}}^T)\check{})^T \mathbf{\Sigma}^{-1} (\ln(\mathbf{R}\bar{\mathbf{R}}^T)\check{})} \tag{3}$$

where Z is the Gaussian normalization constant, ln is the inverse of the exponential map, and $\check{}$ is the inverse linear, skew-symmetric Lie algebra operator.

Unlike the Bingham or Von Mises-Fisher, the Concentrated Gaussian distribution supports the analytical composition of rotations which is a necessary property as we use differential quaternions as intermediate output representation to leverage a residual connection concept (this is justified in Sec. 6.3). A probabilistic multiplication of two rotations $\mathbf{R}_3 = \mathbf{R}_1\mathbf{R}_2$ is expressed as

$$\mathbf{R}_3 = \mathbf{R}_1\mathbf{R}_2 = \exp(\epsilon_1\hat{})\bar{\mathbf{R}}_1 \exp(\epsilon_2\hat{})\bar{\mathbf{R}}_2 \tag{4}$$

without approximation we have

$$\begin{aligned} \mathbf{R}_3 &= \exp(\epsilon_1\hat{})\exp((\bar{\mathbf{R}}_1\epsilon_2)\hat{})\bar{\mathbf{R}}_1\bar{\mathbf{R}}_2 \\ &= \exp(\epsilon_1\hat{})\exp(\epsilon_{12}\hat{})\bar{\mathbf{R}}_3 \end{aligned} \tag{5}$$

using first order Baker-Campbell-Hausdorff approximation we get

$$\mathbf{R}_3 = \exp((\epsilon_1 + \epsilon_{12})\hat{})\bar{\mathbf{R}}_3 \tag{6}$$

$$\mathbf{R}_3 \sim \mathcal{N}_{SO(3)}(\bar{\mathbf{R}}_1\bar{\mathbf{R}}_2, \mathbf{\Sigma}_1 + \bar{\mathbf{R}}_1\mathbf{\Sigma}_2\bar{\mathbf{R}}_1^T) \tag{7}$$

To represent multimodality, we define our output structure as a Concentrated Gaussian Mixture Model in SO(3) ($\mathcal{N}^{\pi}_{SO(3)}$). For $q \sim \mathcal{N}^{\pi}_{SO(3)}(\pi_i, \bar{\mathbf{R}}_i, \mathbf{\Sigma}_i)$, the *p.d.f* is given as

$$p(\mathbf{R} \mid \pi, \bar{\mathbf{R}}, \mathbf{\Sigma}) = \sum_i \pi_i \, \mathcal{N}_{SO(3)}(\bar{\mathbf{R}}_i, \mathbf{\Sigma}_i) \tag{8}$$

where $\pi_i \in \mathbb{R}$ is the mixture coefficient for the i-th $\mathcal{N}_{SO(3)}$ component.

We want to emphasize that the presented concept on differentiable probabilistic (residual) rotations is applicable to a wide range of problems exceeding motion prediction.

Typed Graph Attention. To make use of the information of the human skeleton, the entire model is comprised of two building blocks which preserve and efficiently resemble the skeleton structure. Both modules utilize a Graph Influence Matrix $G \in \mathbb{R}^{N\times N}$ inspired by previous work [26, 33, 36]. Matrix multiplying the Graph Influence Matrix with a Graph State Matrix $\mathbf{x} \in \mathbb{R}^{N\times D_I}$ calculates an element-wise weighted sum of each node's state. This operation is known as *Graph Convolution* [26] or *Graph Attention* [47] where the attention weights are learned instead of inferred from node states.

To allow for model particularities depending on the semantic class S_i of node n_i, we define a typed weight tensor ${}^N\mathbf{W} \in \mathbb{R}^{N\times D_I\times D_O}$ as N stacked weight matrices $\mathbf{W}_S \in \mathbb{R}^{D_I\times D_O}$ where $\mathbf{W}_S$:

$${}^N\mathbf{W} = \begin{bmatrix} \mathbf{W}_{S_0} & \mathbf{W}_{S_1} & \dots & \mathbf{W}_{S_N} \end{bmatrix} \tag{9}$$

We define the multiplication operator $\cdot$ as a batched matrix multiplication between the typed weight tensor ${}^N\mathbf{W}$ and the graph input matrix $\mathbf{x} \in \mathbb{R}^{N\times D_I}$

$$f(\mathbf{x}) = {}^N\mathbf{W} \cdot \mathbf{x} \tag{10}$$

All nodes n_i of the same type S_i share the same weights and all N nodes are processed with a single batched matrix multiplication allowing for efficient learning.

Typed Graph (TG)-Linear: Using both concepts, attention and typed weights, we define the equivalent to a linear fully connected layer in our graph neural network as

$$f(\mathbf{x}) = \mathbf{G}({}^N\mathbf{W} \cdot x) \tag{11}$$

Typed Graph (TG)-GRU: To capture temporal dependencies within the model we introduce the typed graph equivalent to an GRU layer as

$$\begin{aligned} r_t &= \sigma_g(\mathbf{G}_t({}^N\mathbf{W}_r \cdot x_t) + \mathbf{G}_t({}^N\mathbf{U}_r \cdot h_{t-1}) + b_f) \\ z_t &= \sigma_g(\mathbf{G}_t({}^N\mathbf{W}_z \cdot x_t) + \mathbf{G}_t({}^N\mathbf{U}_z \cdot h_{t-1}) + b_f) \\ n_t &= \sigma_g(\mathbf{G}_t({}^N\mathbf{W}_n \cdot x_t) + r_t \circ \mathbf{G}_t({}^N\mathbf{U}_n \cdot h_{t-1}) + b_f) \\ h_t &= (1 - z_t) \circ n_t + z_t \circ h_{t-1} \\ \mathbf{G}_t &= \mathbf{G}_{t-1} + \mathbf{G}_{ta} \end{aligned}$$

where $\mathbf{G}_0$, ${}^N\mathbf{W}$ and ${}^N\mathbf{U}$ are trainable parameters. h is the GRUs state and σ represents an activation function. The input $x \in \mathbb{R}^{N\times D_I}$ holds D_I dimensional information on the N nodes. The Graph Influence Matrix $\mathbf{G}$ is initialized as unit matrix and is optimized during training. For the *TG-GRU* an additional Temporal Additive Graph Influence Matrix $G_{ta} \in \mathbb{R}^{N\times N}$ is initialized as a zero matrix and is optimized to capture the temporal change of influence between nodes over time.

5. Motron

Our model[1] is visualized in Fig. 2. From a high-level perspective, we combine latent discrete variables with probabilistic mixture distribution as output structure to model the diverse nature of human motion while embedding the skeleton graph structure directly into the learning and inference procedure by using only *Typed Graph* components.

This model extends core concepts of probabilistic, multimodal deep learning towards the application of human motion prediction. We call our model *Motron*.

Graph Structure Embodiment. We enable efficient use of graph-structured information by imbuing the architecture from end to end. Thus, the architecture is fully described by the two building blocks *TG-Linear* and *TG-GRU*; there is no fully connected influence between hidden node states. This leads to natural modeling of the information flow, where weights are shared between symmetric joints as well as a reduction in model parameters of about 40%.

Modeling Motion History. Starting from the input representation $\mathbf{x} = \mathbf{s}_{1,\dots,N}^{(t-H:t)} \in \mathbb{R}^{H \times N \times D}$, the model needs to encode a node's current state and its history. To encode the observed history of the node, its current and previous states are fed into a *TG-GRU* network. The final output is encoded to the model's hidden state h by a *TG-Linear* layer.

Explicitly Accounting for Multimodality. *Motron* explicitly handles multimodality by leveraging a probabilistic latent variable architecture. It produces the target distribution $p(\mathbf{y} \mid \mathbf{x})$ by introducing a discrete latent variable z,

$$p(z \mid x) = \frac{1}{N} \sum_i f_{TG-Linear}(h)_i, \tag{12}$$

which encodes high-level latent behavior and allows for $p(\mathbf{y} \mid \mathbf{x})$ to be expressed as

$$p(\mathbf{y} \mid \mathbf{x}) = \sum_z p_\psi(\mathbf{y} \mid \mathbf{x}, z) p_\theta(z \mid \mathbf{x}), \tag{13}$$

where ψ and θ are deep neural network weights that parameterize their respective distributions. Unlike latent variables in probabilistic variational (e.g. (C)VAEs), this latent variable is not explicitly encouraged to learn a representation for the decoder (e.g. using KL-divergence), but implicitly enables the decoder to differentiate between modes.

Producing Motions. The sampled latent variable z is concatenated with the hidden representation vector h and fed into our *TG-GRU* decoder. Each *TG-GRU* cell outputs the parameters $\bar{\mathbf{R}}, \mathbf{\Sigma}$ of a $\mathcal{N}_{SO(3)}$ for each node. Using $\pi_i = p(z = i \mid \mathbf{x})$ we produce the mixture model $\dot{\mathbf{q}} \sim \mathcal{N}^{\pi}_{SO(3)}$ over differential quaternions as an intermediate output distribution.

Thus, z being discrete is necessary as it enables us to rethink Eq. (13) as a $\mathcal{N}^{\pi}_{SO(3)}$ with mixture distribution $p(z \mid x)$. It also aids in interpretability, as one can visualize the high-level behaviors resulting from each z by sampling

[1]All of our source code, trained models, and data can be found online at https://github.com/TUM-AAS/motron-cvpr22.

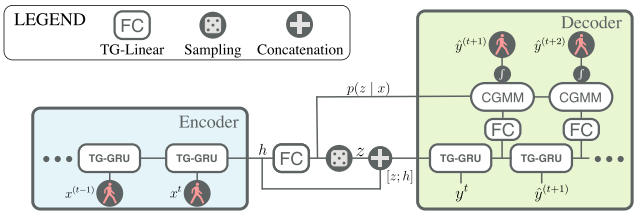


Figure 2. Our network architecture: The encoder abstracts human's historic poses into a hidden representation h using a *TG-GRU*. This representation is used to infer the distribution over the latent variable $p(z \mid \mathbf{x})$, and is fed into the decoder together with latent samples z. The decoder, again, uses a *TG-GRU* to compute the output distribution. Notably, $p(z \mid \mathbf{x})$ is reused as mixing coefficients in the output distribution.

motions (see Fig. 4).

Using the closed composition formula Eq. (6) of the Concentrated Gaussian in SO(3) we can "integrate" the distribution over differential quaternions to the final output distribution over quaternions. The intermediate step is necessary as motion samples are produced by sampling differential quaternions and subsequently "integrating" them to motions. Directly sampling the output distribution would lead to time inconsistent motions. Additionally, using differential output for recurrent layers is known to ease the learning problem and improve convergence [22, 43].

Training the Model. Commonly, learning good representations for probabilistic latent variables is achieved by including the ground truth y as input to the latent layer during training and simultaneously introducing a *Kullback–Leibler Divergence* (KL) loss term to *squeeze* out the dependency on y during the training process [19, 43]. When using the CVAE framework, these competing conditions can lead to unstable training behavior and the collapse of the latent distribution (KL divergence towards zero). In contrast, we do not input the ground truth but give the model the option to use the latent capacity $p_\theta(z \mid \mathbf{x})$ to maximize Eq. (13) in Eq. (14). Formally, we aim to solve

$$\max_{\theta,\psi} \sum_{i=1}^{N} \mathbb{E}_{z \sim p_\theta(\cdot \mid \mathbf{x}_i)} \left[\log p_\psi(\mathbf{y}_i \mid \mathbf{x}_i, z) \right] \tag{14}$$

Notably, no reparameterization trick, commonly needed for training probabilistic latent variable models [24, 25] is used to backpropagate through the categorical latent variable z as it is not sampled during training time. Instead, Eq. (14) is directly computed since the latent space has only $|Z|$ discrete elements. (For an in depth discussion see Appendix A)

Output Configurations. Based on the desired use case, *Motron* can produce many different outputs. The main four are outlined below.

1. *Distribution*: Due to the use of a discrete latent variable and probabilistic output structure, the model can provide an analytic output distribution by directly computing Eq. (13). This parametric $\mathcal{N}^{\pi}_{SO(3)}$ distribution entails the complete information inferred by the model.

2. *Sampled*: The model's sampled output, where z and y are sampled sequentially according to

$$z \sim p_\theta(z \mid \mathbf{x}), \qquad \mathbf{y} \sim p_\psi(\mathbf{y} \mid \mathbf{x}, z). \tag{15}$$

3. *Most Likely Mode (ML-Mode)*: The model's deterministic and most-likely single output. The high-level latent behavior mode and output trajectory are the modes of their respective distributions, where

$$\begin{aligned} z_{\text{mode}} &= \arg\max_z p_\theta(z \mid \mathbf{x}), \\ \mathbf{y} &= \arg\max_{\mathbf{y}} p_\psi(\mathbf{y} \mid \mathbf{x}, z_{\text{mode}}). \end{aligned} \tag{16}$$

4. *Weighted Mean (W-Mean)* The mean of all latent modes weighted by their probability. Following [37], it is given as the normalized largest eigenvector of QQ^T where Q is a matrix of stacked quaternion column vectors, where

$$\begin{aligned} \mathbf{q}_n = \mathbf{y}_n &= \arg\max_{\mathbf{y}} p(\mathbf{y} \mid \mathbf{x}, z = n), \\ Q &= [\pi_1 \mathbf{q}_1, ..., \pi_{|Z|} \mathbf{q}_{|Z|}]. \end{aligned} \tag{17}$$

6. Experiments

While desiderata (III) and (V) in Sec. 1 are explicitly fulfilled by the embedded human structure and the use of $\mathcal{N}^{\pi}_{SO(3)}$ as output distribution, we conduct both quantitative and qualitative experiments to show that our method also succeeds in the remaining desiderata.

We, therefore, structure our experiments as follows: First, we show that *Motron* performs best in capturing humans' probabilistic and diverse nature (II) by introducing a new probabilistic metric to the field of human motion prediction. Secondly, we highlight the learned multimodal structure (I) by evaluating deterministic outputs of high likelihood modes. Later we also give a visualization of how these modes manifest in distinct motions. Finally, we show that our approach can handle incomplete data (IV) in the form of occluded joints.

Datasets. We provide quantitative experimentation results on two datasets; namely the *Human 3.6 Million* (H3.6M) [20] dataset and the *Archive of Motion Capture as Surface Shapes* (AMASS) [34]. AMASS is a unified collection of 18 motion capture datasets totaling 13944 motion-sequences from 460 subjects performing a large variety of actions. In comparison, the H3.6M dataset consists of 240 sequences from 8 subjects performing 15 actions each.

6.1. Probabilistic Evaluation

Metrics. Probabilistic approaches have been compared on the basis of a variety of metrics in position space, most prominent are Best-of-N metrics where $N = 50$ motions are sampled from the model; the best metric value of these is reported. Such metrics, however, only present limited insights on a model's output distribution as only a single motion of an arbitrary number N is evaluated. To fully assess an algorithm's probabilistic capabilities, we propose an alternative evaluation methodology for probabilistic algorithms where we measure the ability to accurately capture and reproduce the underlying uncertainty distribution of motions. To this point we adopt the *KDE-NLL* metric [21] to assess the method's Negative Log-Likelihood (NLL) by fitting a probability distribution, using Kernel Density Estimate (KDE) [39], to output samples. Although Motron can compute its own log-likelihood, we apply the same evaluation methodology to maintain a directly comparable performance measure. As the NLL is unbounded, we clip it to a maximum value of 20 ($\sim 2 * 10^{-7}\%$) in order to prevent single outliers from dominating. Still, to be comparable, we additionally follow the evaluation methodology of Yuan *et al*. [53]: We report the *Average Pairwise Distance* (APD) as a measure of sample diversity as well as the Best-of-N metrics *Average Displacement Error* (ADE), and *Final Displacement Error* (FDE) as measures of quality. Further, we report their Multi-Modal ADE and FDE metrics (MMADE and MMFDE). "Similar" motions are grouped by using an arbitrary distance threshold at $t = 0$ and the average metric over all these grouped motions is reported. As close poses at a single instance can, however, belong to entirely different motions (Appendix C), the KDE-NLL reports a more holistic representation of a model's probabilistic capabilities.

Probabilistic Baselines. We focus on the current state-of-the-art algorithm (1) *DLow* [53] to compare our desiderata side to side. *DLow* uses an adapted VAE algorithm to generate samples without collapsing to a single mode. For the standard probabilistic experiment methodology, we further report methods based on CVAEs: (2) *Pose-Knows* [48] and (3) *MT-VAE* [51]; GAN based (4) *DeLiGAN* [18] and diversity promoting methods (5) *Best-of-Many* [5], (6) *GM-VAE* [9], and (7) *DSF* [52]. We were not able to compare to *LCP-VAE* [2] for a lack of open accessible source code.

Evaluation Methodology. In order to compare to other probabilistic methods, which output motions solely in position space, we use the forward kinematic of the respective test subject to convert our joint configuration samples to joint positions. Predicting in configuration space and using human's forward kinematic to produce motions in position space, ensures our motions to be kinematic feasible. However, they bring a disadvantage compared to approaches directly outputting joints positions, common for probabilistic approaches, as they are not constrained by a rigid bone structure (Appendix D). For the KDE-NLL metric, we sample $N = 1000$ motions from each method and fit a KDE for each future timestep. For the $N = 50$ metrics (APD, ADE, FDE, MMADE, MMFDE) we use 50 motions, sampled from our intermediate output distribution over $\dot{q}$ and "integrated"; these motions are transformed into position space using the respective subjects forward kinematics. We train the model on 0.5 seconds of history and predict a two second horizon. Generally, we can dynamically change the prediction horizon online thanks to the flexible decoder structure during inference.

Results. Fig. 3 shows that *Motron* clearly outperforms the current state-of-the-art algorithm *DLow* in representing the underlying motion distribution of the human sub-

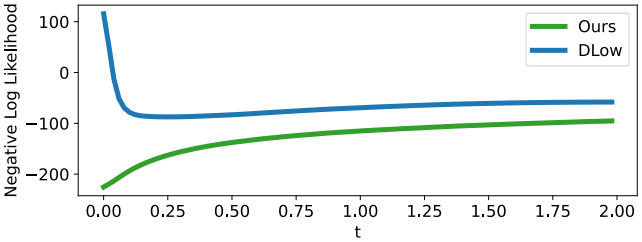


Figure 3. Negative Log Likelihood of the ground truth in the fitted KDE distribution of samples from each model. Lower is better. Samples from *DLow* are over-confidently wrong for early prediction timesteps.

	1 s			2 s		
	APD	ADE	FDE	APD	ADE	FDE
DSF [52]	-	-	-	9.330	0.493	0.592
DeLiGAN [18]	-	-	-	6.509	0.483	0.534
GMVAE [9]	-	-	-	6.769	0.461	0.555
Best-of-Many [5]	-	-	-	6.265	0.448	0.533
MT-VAE [51]	-	-	-	0.403	0.457	0.595
Pose-Knows [48]	-	-	-	6.723	0.461	0.560
DLow [53]	5.180	0.305	0.419	11.741	0.425	0.518
Ours	3.453	**0.252**	**0.350**	7.168	**0.375**	**0.488**

Table 1. Best of $N = 50$ evaluation against probabilistic algorithms on a prediction horizon of one and two seconds on H3.6M dataset.

ject and therefore having a lower NLL over all prediction timesteps. This result is supported by the Best-of-N metrics over one and two seconds in Tab. 1. We show significantly better results than *DLow* on ADE and FDE. Notably, this is achieved while using less diverse motion samples (lower APD) indicating that our samples are concentrated around likely motions. MMADE and MMFDE values for different thresholds are presented in Fig. 6 in Appendix C. For smaller thresholds we outperform *DLow* while for higher thresholds, where possibly uncorrelated motions are evaluated together as displayed in Fig. 8 in Appendix C, we unsurprisingly achieve lower scores as our approach does not over-diversify its predictions.

6.2. Deterministic Evaluation

Metrics. We report the *Mean Angle Error* (MAE-L2) as the Euclidean distance of the stacked (ZYX-)Euler angles as well as the *Mean Per Joint Position Error* (MPJPE) [20] which is calculated using the human's skeleton forward kinematic. Those are the standard evaluation metrics in deterministic motion prediction [23, 33, 35, 36, 40]. To better understand the influence of the learned latent multimodality we apply an *Best-of-N* evaluation for the MAE-L2 and MPJPE metric. Here we report the value for the motion with the best average metric value over all prediction timesteps originating from the N modes' means with the highest probability $p(z \mid x)$.

Deterministic Baselines. We evaluate against the following deterministic approaches: (1) *Zero Velocity*: All joints keep their current state at prediction time throughout the entire prediction horizon. (2) *GRU sup.* [38]: Simple encoder-decoder structure using GRUs for variable history and prediction horizon and exponential maps as data representation. (3) *Quaternet* [40]: Encoder-Decoder architecture using GRUs and quaternion data representation. (4) *HistRepItself* [35]: Transformer [46] encoder and fixed prediction horizon graph convolution decoder. Data is preprocessed using the *Discrete Cosine Transform* (DCT) on the time dimension and the model predicts a residual before the output is transformed back using the inverse DCT. (5) *ST-Transformer* [1]: Decoupled temporal and spatial self-attention. For the H3.6M dataset we state the reported values of (2) - (3) and re-run the evaluation of (4) as they originally did not account for 2π angle discontinuity. For the AMASS dataset we re-train (4) *HistRepItself* to match our test split. We were not able to compare to some other methods for a lack of open source code (*AGED* [17]) or their computational complexity (*DMGNN* [33] - 62 Million parameter)

Evaluation Methodology. It has become common to benchmark on 8 fixed sequences per action of a single test subject on the H3.6M dataset. This has been shown to be un-representative [40]. Thus, we report results on 8 [33, 35, 38, 40] (see Appendix F) and 256 [35, 40] samples per action on the H3.6M data to be comparable to past methods. For the AMASS dataset, we report results on the official test split, consisting of the *Transitions* and *SSM* dataset. While prior authors [35] have argued that the *Transitions* dataset is not suitable for evaluating prediction algorithms for their change of action within sequences, we argue that such behavior can happen in real-world applications. We subsample the H3.6M dataset to 25 HZ and the AMASS dataset to 20 HZ as most of the included datasets have been recorded with a framerate divisible by 20 but not by 25. When comparing to *ST-Transformer* [1] (Tab. 4) we follow their evaluation methodology. For the H3.6M dataset, we report metric values previously published on both 8 and 256 samples per action. As the AMASS dataset has been released recently, there are not many published results, yet. Thus, we retrain the current best state-of-the-art algorithm *HistRepItself* and re-train it on the official test split. We train the model on two seconds of history and predict one second into the future.

Results. We evaluate our approach on common single sample metrics against fully deterministic approaches. Tab. 2 summarizes the results on the H3.6M dataset. Even though we don't explicitly optimize for a deterministic output, our Weighted-Mean output outperforms all other state-of-the-art algorithms.

We want to point out that we introduced the W-Mean output configuration solely for these metrics commonly used in deterministic evaluation. This allows us to point to the shortcomings of both, deterministic algorithms and their corresponding evaluation metrics: They produce motions which represent the average of all likely motions given the motion history. This average motion, however, may represent a unlikely or even infeasible motion. This (unintentional) behavior is followed by our W-Mean output configuration. Thus, we want to emphasize that while we outperform other algorithms on specific metrics we advise against

	MAE (L2)							
milliseconds	80	160	320	400	560	720	880	1000
Zero Vel.	0.40	0.70	1.11	1.25	1.46	1.63	1.76	1.84
GRU sup. [38]	0.43	0.74	1.15	1.30	-	-	-	-
Quarternet [40]	0.37	0.62	1.00	1.14	-	-	-	
HistRepItself [35]	**0.28**	0.52	0.88	1.02	1.23	**1.40**	1.55	1.64
Ours W-Mean	**0.28**	**0.51**	**0.87**	**1.01**	**1.22**	**1.40**	**1.54**	**1.63**
Ours ML-Mode	0.28	0.51	0.88	1.02	1.24	1.42	1.58	1.67
Ours Bo3-Modes	0.28	0.50	0.85	0.97	1.16	1.31	1.45	1.54
Ours Bo5-Modes	0.28	0.51	0.84	0.96	1.13	1.28	1.42	1.51

Table 2. Average angle error on 256 samples per action on the H3.6M test dataset. A break down by actions and the results on the MPJPE metric can be found in Appendix F.

	MAE (L2)					
milliseconds	100	200	400	600	800	1000
Zero Vel.	0.73	1.20	1.60	1.73	1.76	1.76
HistRepItself [35]	0.45	0.78	1.06	**1.17**	**1.27**	**1.33**
Ours W-Mean	**0.42**	**0.76**	**1.05**	1.19	**1.27**	**1.33**
Ours ML-Mode	0.42	0.76	1.08	1.22	1.31	1.38
Ours Bo3-Modes	0.42	0.74	1.01	1.12	1.20	1.28
Ours Bo5-Modes	0.42	0.74	0.99	1.09	1.17	1.27

Table 3. Average angle error on 10,000 samples from the AMASS test set. The MPJPE metric can be found in Appendix F.

	MAE (L2)			
milliseconds	100	200	300	400
ST-Transformer [1]	0.178	0.291	0.395	0.490
Ours W-Mean	**0.147**	**0.243**	**0.335**	**0.420**

Table 4. Average angle error using the evaluation procedure in [1].

using the W-Mean output configuration for actual applications.

The lower rows of Tab. 2 and Tab. 3 supports that our approach captures multimodality by committing to a specific motion per mode; thereby attributing appropriate probability mass to less likely motions. As such, the most likely deterministic mode (ML-Mode) commits to the mode which is best explained by the data. However, on average it accumulates a higher error compared to W-Mean as it is expected to represent a "wrong" motion with probability $p = 1 - \max_i(\pi_i)$ (see Sec. 5). Looking at the set of likely deterministic mode outputs (BoN in Tab. 2 and Tab. 3), it becomes clear that one of the motion modes performs exceptionally better than the mean output of deterministic regressors. This behavior is further visualized for a single example in Sec. 6.4.

Notably, *Motron* can capture multimodality and reason probabilistically about humans' future motion while being computationally more efficient then a deterministic regressor. With 1.7 Million parameters we need **half** the computational power than *HistRepItself* (3.4M parameters) and are significantly more efficient compared to the current state-of-the-art algorithm *DLow* with 7.3 Million parameters. Other graph based models, such as [33], even use 62 Million parameters.

6.3. Ablation Study

In this section, we will show the influence of our contributions on the model's performance as well as justify our design choices quantitatively.

	NLL			MAE (L2)	
milliseconds	400	1000	$\sum$	400	1000
$\|Z\| = 1$	-160.77	-106.10	-4032.58	1.05	1.68
$\|Z\| = 2$	-173.21	-117.20	-4320.73	1.02	1.63
$\|Z\| = 3$	-176.25	-121.86	-4405.46	1.02	1.65
$\|Z\| = 4$	-176.74	**-122.03**	-4418.98	**1.01**	**1.63**
$\|Z\| = 5$	**-177.01**	-122.02	**-4432.40**	**1.01**	**1.63**
$\|Z\| = 6$	-174.50	-117.70	-4340.94	1.01	1.64

Table 5. Negative Log Likelihood (NLL) and MAE (L2) performance using different number of latent modes on H3.6M dataset.

	NLL			MAE (L2)	
milliseconds	400	1000	$\sum$	400	1000
No Typed Graph	-170.28	-112.35	-4264.17	1.07	1.73
One-Hot	-174.78	-119.33	-4372.70	1.04	1.67
Gaussian Mixture Model	-158.56	-102.24	-3879.12	1.06	1.69
Bingham	-162.58	-107.52	-3983.10	1.07	1.67
Latent Grad. Flow	-174.68	-119.99	-4374.77	1.02	1.64
Full	**-177.01**	**-122.02**	**-4432.40**	**1.01**	**1.63**

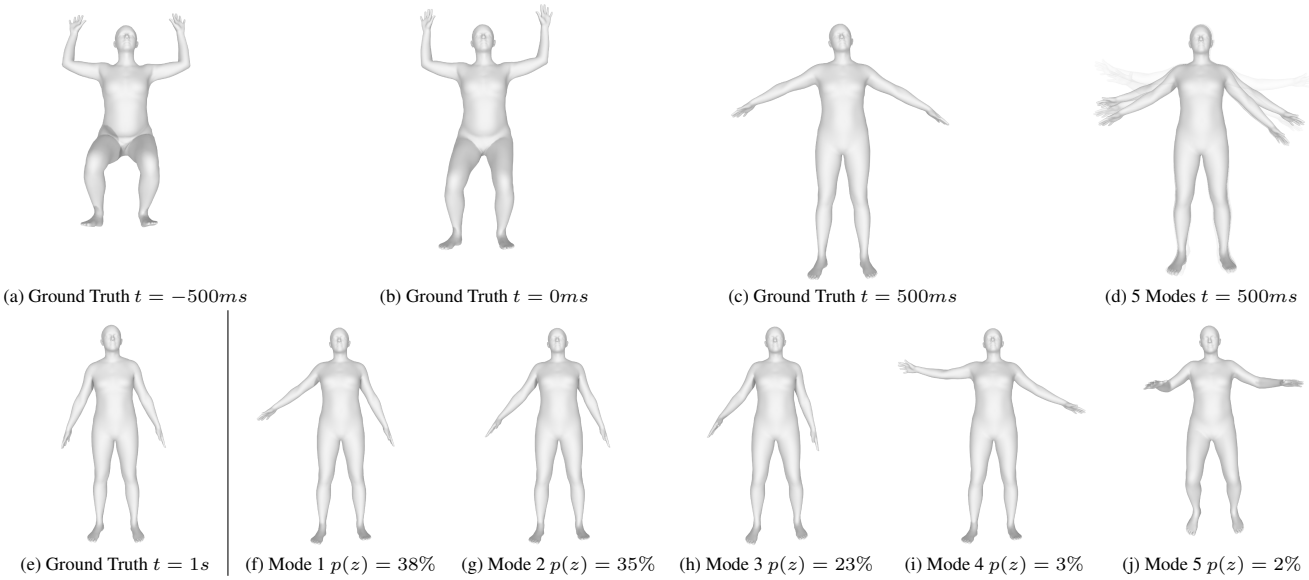
Table 6. Negative Log Likelihood (NLL) and MAE (L2) performance on ablated model structures on H3.6M dataset.

Number of latent modes. We ablate the number of latent discrete states which manifest as motion modes in the model's output (Tab. 5). Four and five modes show the best performance. We use five modes for our approach to give the model more expressiveness when necessary.

Influence of contributions. To show the influence of our contributions on the model's performance we ablate them individually in Tab. 6. We first remove the *Typed Graph* weight sharing scheme and subsequently replace it with One-Hot encoded type information which can recover some performance. Next, we replace the Concentrated Gaussian distribution in SO(3) with a standard Multivariate Normal (Gaussian) Mixture Model (GMM) and subsequently with a Bingham distribution. Unlike our $\mathcal{N}_{SO(3)}$, the MVN is not designed to handle rotations and their special characteristics. It, therefore, has worse results in all metrics. The Bingham distribution, in contrast, is designed for rotations but does not support the composition of rotations. Thus, we can not use differential quaternions as intermediate output making the learning task more complex. Also, samples from a Bingham can only be approximated using computationally and memory expensive algorithms (e.g. Metropolis-Hasting). Finally, we enable gradient flow through the latent variable (see Appendix A and Sec. 5), which has a minor negative impact.

6.4. Qualitative Results

Fig. 4 shows the capability of our method to capture the multimodal nature of human motions. Displayed is an exemplary motion where the human lands from a jump and rapidly pulls his arms down. In the beginning $t = 500ms$, the different latent modes capture different possible speeds of the downwards arm movement. While the most likely mode anticipates a slower downwards movement than performed, the second most likely mode captures the true motion closely. Further, less likely modes capture even faster motions as well as movements where the arms are pulled more in front of the body. Notably, reasonably small uncer-



(a) Ground Truth $t = -500ms$ (b) Ground Truth $t = 0ms$ (c) Ground Truth $t = 500ms$ (d) 5 Modes $t = 500ms$

(e) Ground Truth $t = 1s$ (f) Mode 1 $p(z) = 38\%$ (g) Mode 2 $p(z) = 35\%$ (h) Mode 3 $p(z) = 23\%$ (i) Mode 4 $p(z) = 3\%$ (j) Mode 5 $p(z) = 2\%$

Figure 4. Qualitative visualization of the prediction distribution on a single sample from the AMASS dataset. The human lands from a jump and pulls both arms downwards. (a)-(c), (e) Ground Truth poses. (d) 5 modes of our prediction distribution. Each mode is weighted by its probability where opacity indicates high confidence in a particular mode. (f)-(j) Mean of each of the five output modes at $t = 1s$.

tainty is presented by the model for other joints. Towards the end $t = 1000ms$, the expressiveness and multimodality of the modes can be experienced as, for example, Mode 5 captures the possibility of a consecutive second jump.

Another important quality for robotic applications is the ability to work with imperfect data. To simulate occlusions during training we apply *Node Dropout*. For the occluded nodes, we set a random number of continuous states leading to $t = 0$ to the neutral quaternion. The unique capabilities of our model here are shown in Fig. 5. In this instance, we artificially occlude all joint's data of the left leg. This leads to high variance but reasonable sampled predictions during early timesteps. More importantly, the model can understand and output its uncertainty: The closed form standard deviation of the parametric $\mathcal{N}^{\pi}_{SO(3)}$ output distribution is adequately higher compared to the distribution with perfect data. For increasing prediction time, the model uses the influence between nodes learned in the *Typed Graph* layers to produce reasonable motions even for the occluded nodes and adjusts its relative confidence reasonably.

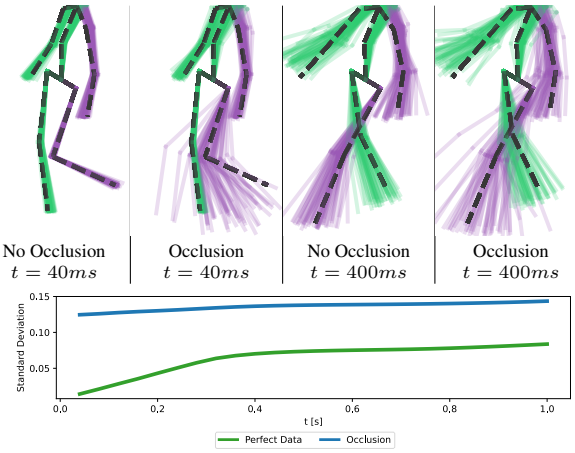


Figure 5. Handling of imperfect data. All joints of the left leg are artificially occluded. *Top*: Visualization of prediction samples with and without occlusion. *Bottom*: Mean standard deviation of parametric output distribution of left hip and knee. The occluded data is addressed by the model adjusting its own uncertainty.

7. Conclusion

In this work, we present *Motron*, a probabilistic human motion forecasting approach which uniquely provides the information plethora of a probabilistic approach and the accuracy of a deterministic model. Its predictions respect rigid skeleton constraints, all while producing full parametric motion distributions, which can be especially useful in downstream robotic applications. It achieves state-of-the-art prediction performance in a variety of metrics on standard and new real-world human motion datasets. Further, to the best of the authors' knowledge, it is the first method that demonstrates its ability to deal with occluded data while reasoning about its own uncertainty.

Limitations. The approach is yet limited by the upstream data provider. While the motion capture system utilized to record the datasets used here, provides the required accuracy to calculate the joint rotations via inverse kinematics, the authors anticipate this being a challenge when relying on vision-based algorithms for human poses detection. Further, our approach's tendency to less diverse predictions can become problematic with regards to new unseen behaviors where our approach would be overly confident.

Future Directions include incorporating *Motron*'s human behavior predictions in downstream robotic planning, decision making, and control frameworks, as well as exploring options to tightly couple upstream vision algorithms.

References

[1] Emre Aksan, Manuel Kaufmann, Peng Cao, and Otmar Hilliges, "A Spatio-temporal Transformer for 3D Human Motion Prediction," in *2021 Int. Conf. on 3D Vision (3DV)*, 2021.

[2] Sadegh Aliakbarian, Fatemeh Saleh, Lars Petersson, Stephen Gould, *et al.*, "Contextually Plausible and Diverse 3D Human Motion Prediction," in *2021 IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, 2021.

[3] Timothy D Barfoot, *State Estimation for Robotics*. 2021.

[4] Emad Barsoum, John Kender, and Zicheng Liu, "HP-GAN: Probabilistic 3D Human Motion Prediction via GAN," in *2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2018.

[5] Apratim Bhattacharyya, Bernt Schiele, and Mario Fritz, "Accurate and Diverse Sampling of Sequences Based on a "Best of Many" Sample Objective," in *2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, 2018.

[6] Christopher Bingham, "An antipodally symmetric distribution on the sphere," *The Annals of Statistics*, 1974.

[7] Eli Bingham, Jonathan P. Chen, Martin Jankowiak, Fritz Obermeyer, *et al.*, "Pyro: Deep Universal Probabilistic Programming," *Journal of Machine Learning Research*, 2018.

[8] Christopher M Bishop, "Mixture Density Networks," Tech. Rep., 1994.

[9] Nat Dilokthanakul, Pedro A. M. Mediano, Marta Garnelo, Matthew C. H. Lee, *et al.*, "Deep Unsupervised Clustering with Gaussian Mixture Variational Autoencoders," in *Int. Conf. on Learning Representations*, 2017.

[10] Ronald Fisher, "Dispersion on a Sphere," *Proceedings of the Royal Society of London. Series A*, 1953.

[11] Katerina Fragkiadaki, Sergey Levine, Panna Felsen, and Jitendra Malik, "Recurrent Network Models for Human Dynamics," in *2015 IEEE Int. Conf. on Computer Vision (ICCV)*, 2015.

[12] Partha Ghosh, Jie Song, Emre Aksan, and Otmar Hilliges, "Learning Human Motion Models for Long-Term Predictions," in *2017 Int. Conf. on 3D Vision (3DV)*, 2017.

[13] Igor Gilitschenski, Gerhard Kurz, Simon J. Julier, and Uwe D. Hanebeck, "Efficient Bingham filtering based on saddlepoint approximations," in *2014 Int. Conf. on Multisensor Fusion and Information Integration for Intelligent Systems (MFI)*, 2014.

[14] Igor Gilitschenski, Roshni Sahoo, Wilko Schwarting, Alexander Amini, *et al.*, "Deep Orientation Uncertainty Learning Based on a Bingham Loss," in *Int. Conf. on Learning Representations (ICLR)*, 2020.

[15] Jared Glover, "The Quaternion Bingham Distribution, Detection, and Dynamic Manipulation," Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, 2014.

[16] F Sebastian Grassia, "Practical Parameterization of Rotations Using the Exponential Map," Tech. Rep., 1998.

[17] Liang-Yan Gui, Yu-Xiong Wang, Xiaodan Liang, and José M. F. Moura, "Adversarial Geometry-Aware Human Motion Prediction," in *Proceedings of the European Conf. on Computer Vision (ECCV)*, 2018.

[18] Swaminathan Gurumurthy, Ravi Kiran Sarvadevabhatla, and R. Venkatesh Babu, "DeLiGAN: Generative Adversarial Networks for Diverse and Limited Data," in *2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[19] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, *et al.*, "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework," in *Int. Conf. on Learning Representations*, 2016.

[20] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu, "Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2014.

[21] Boris Ivanovic and Marco Pavone, "The Trajectron: Probabilistic Multi-Agent Trajectory Modeling With Dynamic Spatiotemporal Graphs," in *2019 IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, 2019.

[22] Boris Ivanovic, Edward Schmerling, Karen Leung, and Marco Pavone, "Generative Modeling of Multimodal Multi-Human Behavior," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2018.

[23] Ashesh Jain, Amir R. Zamir, Silvio Savarese, and Ashutosh Saxena, "Structural-RNN: Deep Learning on Spatio-Temporal Graphs," in *2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[24] Eric Jang, Shixiang Gu, and Ben Poole, "Categorical Reparameterization with Gumbel-Softmax," in *Int. Conf. on Learning Representations*, 2017.

[25] Diederik P Kingma and Max Welling, "Auto-Encoding Variational Bayes," in *arXiv preprint*, 2013.

[26] Thomas N. Kipf and Max Welling, "Semi-Supervised Classification with Graph Convolutional Networks," in *Int. Conf. on Learning Representations*, 2017.

[27] Vineet Kosaraju, Amir Sadeghian, Roberto Martín-Martín, Ian Reid, *et al.*, "Social-BiGAT: Multimodal Trajectory Forecasting using Bicycle-GAN and Graph Attention Networks," in *Advances in Neural Information Processing Systems*, 2019.

[28] Jogendra Nath Kundu, Maharshi Gor, and R. Venkatesh Babu, "BiHMP-GAN: Bidirectional 3D Human Motion Prediction GAN," in *AAAI Conf. on Artificial Intelligence*, 2018.

[29] Gerhard Kurz, Igor Gilitschenski, and Uwe D Hanebeck, "Efficient Evaluation of the Probability Density Function of a Wrapped Normal Distribution," in *IEEE ISIF Workshop on Sensor Data Fusion*, 2014.

[30] Gerhard Kurz, Igor Gilitschenski, Florian Pfaff, Lukas Drude, *et al.*, "Statistics and Filtering Using libDirectional," *Journal of Statistical Software Directional*, 2017.

[31] Muriel Lang, "Approximation of probability density functions on the Euclidean group parametrized by dual quaternions," Ph.D. dissertation.

[32] Andreas M. Lehrmann, Peter V. Gehler, and Sebastian Nowozin, "Efficient Nonlinear Markov Models for Human Motion," in *2014 IEEE Conf. on Computer Vision and Pattern Recognition*, 2014.

[33] Maosen Li, Siheng Chen, Yangheng Zhao, Ya Zhang, *et al.*, "Dynamic Multiscale Graph Neural Networks for 3D Skeleton Based Human Motion Prediction," in *2020 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[34] Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, *et al.*, "AMASS: Archive of Motion Capture as Surface Shapes," in *Int. Conf. on Computer Vision*, 2019.

[35] Wei Mao, Miaomiao Liu, and Mathieu Salzmann, "History Repeats Itself: Human Motion Prediction via Motion Attention," in *European Conf. on Computer Vision*, 2020.

[36] Wei Mao, Miaomiao Liu, Mathieu Salzmann, and Hongdong Li, "Learning Trajectory Dependencies for Human Motion Prediction," in *2019 IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, 2019.

[37] F. Landis Markley, Yang Cheng, John Crassidis, and Yaakov Oshman, "Averaging Quaternions," in *Journal of Guidance, Control, and Dynamics*, 2007.

[38] Julieta Martinez, Michael J. Black, and Javier Romero, "On Human Motion Prediction Using Recurrent Neural Networks," in *2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[39] Emanuel Parzen, "On Estimation of a Probability Density Function and Mode," *Ann. Math. Statist.*, 1962.

[40] Dario Pavllo, Christoph Feichtenhofer, Michael Auli, and David Grangier, "Modeling Human Motion with Quaternion-based Neural Networks," *Int. Journal of Computer Vision*, 2019.

[41] Vladimir Pavlovic, James M Rehg, and John Maccormick, "Learning Switching Linear Models of Human Motion," in *Advances in Neural Information Processing Systems*, 2000.

[42] Valentin Peretroukhin, Matthew Giamou, David M. Rosen, W. Nicholas Greene, *et al.*, "A Smooth Representation of Belief over SO(3) for Deep Rotation Learning with Uncertainty," in *Proceedings of Robotics: Science and Systems (RSS)*, 2020.

[43] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone, "Trajectron++: Dynamically-Feasible Trajectory Forecasting with Heterogeneous Data," in *European Conf. on Computer Vision (ECCV)*, 2020.

[44] Yichuan Charlie Tang and Ruslan Salakhutdinov, "Multiple Futures Prediction," in *Advances in neural information processing systems*, 2019.

[45] Yongyi Tang, Lin Ma, Wei Liu, and Wei-Shi Zheng, "Long-Term Human Motion Prediction by Modeling Motion Context and Enhancing Motion Dynamics," in *Int. Joint Conf. on Artificial Intelligence*, 2018.

[46] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, *et al.*, "Attention Is All You Need," in *Advances in Neural Information Processing Systems*, 2017.

[47] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, *et al.*, "Graph Attention Networks," in *Int. Conf. on Learning Representations*, 2017.

[48] Jacob Walker, Kenneth Marino, Abhinav Gupta, and Martial Hebert, "The Pose Knows: Video Forecasting by Generating Pose Futures," in *Int. Conf. on Computer Vision*, 2017.

[49] Jack M. Wang, David J. Fleet, and Aaron Hertzmann, "Gaussian process dynamical models for human motion," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2008.

[50] Tianlong Wu, Feng Chen, and Yun Wan, "Graph Attention LSTM Network: A New Model for Traffic Flow Forecasting," in *2018 5th Int. Conf. on Information Science and Control Engineering (ICISCE)*, 2018.

[51] Xinchen Yan, Akash Rastogi, Ruben Villegas, Kalyan Sunkavalli, *et al.*, "MT-VAE: Learning Motion Transformations to Generate Multimodal Human Dynamics," in *European Conf. on Computer Vision*, 2018.

[52] Ye Yuan and Kris Kitani, "Diverse Trajectory Forecasting with Determinantal Point Processes," in *Int. Conf. on Learning Representations*, 2020.

[53] Ye Yuan and Kris Kitani, "DLow: Diversifying Latent Flows for Diverse Human Motion Prediction," in *Proceedings of the European Conf. on Computer Vision (ECCV)*, 2020.

[54] Yan Zhang, Michael J. Black, and Siyu Tang, "We are More than Our Joints: Predicting how 3D Bodies Move," in *Proceedings of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, 2021.

[55] Jie Zhou, Ganqu Cui, Zhengyan Zhang, Cheng Yang, *et al.*, "Graph Neural Networks: A Review of Methods and Applications," *AI Open*, 2020.