2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Madrid, Spain, October 1-5, 2018

Teaching Robots to Predict Human Motion

Liang-Yan Gui[1], Kevin Zhang[2], Yu-Xiong Wang[2], Xiaodan Liang[3], José M. F. Moura[1], Manuela Veloso[3]

Abstract— **Teaching a robot to predict and mimic how a human moves or acts in the near future by observing a series of historical human movements is a crucial first step in human-robot interaction and collaboration. In this paper, we instrument a robot with such a prediction ability by leveraging recent deep learning and computer vision techniques. First, our system takes images from the robot camera as input to produce the corresponding human skeleton based on real-time human pose estimation obtained with the OpenPose library. Then, conditioning on this historical sequence, the robot forecasts plausible motion through a motion predictor, generating a corresponding demonstration.**

Because of a lack of high-level fidelity validation, existing forecasting algorithms suffer from error accumulation and inaccurate prediction. Inspired by generative adversarial networks (GANs), we introduce a global discriminator that examines whether the predicted sequence is smooth and realistic. Our resulting *motion GAN* model achieves superior prediction performance to state-of-the-art approaches when evaluated on the standard H3.6M dataset. Based on this motion GAN model, the robot demonstrates its ability to replay the predicted motion in a human-like manner when interacting with a person.

I. Introduction

Consider the following scenario: a robot is dancing with a human. In a perfect dancing show, the robot not only recognizes but also anticipates human actions, accurately predicting limbs' pose and position, so that it can interact appropriately and seamlessly. The first step towards this ambitious goal is for the robot to *predict and demonstrate human motion* by observing human activities. More specifically, as illustrated in Figure 1, while a person performs certain action, the robot watches and mimics the person's movements. After the person stops, the robot predicts plausible future motion of that person and generates a corresponding demonstration.

A core component in such a human-robot interaction and collaboration [1], [2], [3] system is human motion prediction that forecasts how a human moves or acts in the near future by conditioning on a series of historical movements [4], [5], [6], [7]. In addition, human motion prediction has wide application potential in a variety of robotic vision tasks, including action anticipation [8], [9], motion generation [10], and autonomous driving systems [11].

Predicting plausible human motion for diverse actions, however, is a challenging yet under-explored problem, because of the uncertainty of human conscious movements and the difficulty of modeling motion dynamics. Traditional approaches focus on bilinear spatio-temporal basis models [12], hidden Markov models [13], Gaussian process latent variable models [14], linear dynamic models [15], and restricted Boltzmann machines [16], [17]. More recently, driven by the advances of deep learning architectures and large-scale public datasets, various deep learning based techniques have been proposed and have significantly pushed the state of the art [4], [5], [6], [7]. They formulate the task as a sequence-to-sequence problem and solve it by using recurrent neural networks (RNNs) to capture the underlying temporal dependencies in the sequential data. Despite their extensive efforts on exploring recurrent encoder-decoder architectures (*e.g.*, encoder-recurrent-decoder (ERD) [4] and residual [6] architectures), they can only predict periodic actions well (*e.g.*, walking) and show unsatisfactory performance on aperiodic actions (*e.g.*, discussion), due to error accumulation.

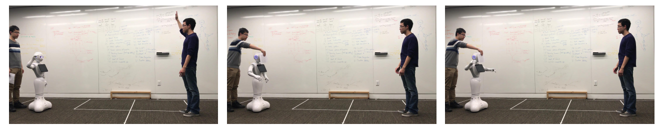

Fig. 1: Human motion prediction in human-robot interaction and collaboration. **Left:** while a person is standing in front of a robot and performing the "greeting" action, the robot is observing and mimicking the person. **Middle:** the robot's eyes are blinded with a sheet of paper, indicating no sensory inputs. **Right:** the robot is demonstrating the predicted "greeting" motion and interacting with the person.

In this work, we aim to address human-like motion prediction that ensures temporal coherence and fidelity of the predicted motion and that can be deployed on the robot for its interaction with humans. To achieve this, we propose a novel *motion GAN* model that learns to validate the motion prediction generated by the encoder-decoder network through *a global discriminator in an adversarial manner*.

Generative adversarial networks (GANs) [18] have shown great progress in image generation and video sequence generation by jointly optimizing a generator and a discriminator in a competitive game, where the discriminator aims to distinguish the generated samples from the samples of the training set and the generator tries to fool the discriminator. In the spirit of GANs, we cast the encoder-decoder network based predictor as a generator and introduce a discriminator to validate the fidelity of the predicted motion sequence. The discriminator aims to examine whether the generated motion sequence is human-like and smooth overall by comparing the

[1]Liang-Yan Gui and José M. F. Moura are with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, USA {lgui, moura}@andrew.cmu.edu

[2]Kevin Zhang and Yu-Xiong Wang are with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA {klz1, yuxiongw}@cs.cmu.edu

[3]Xiaodan Liang and Manuela Veloso are with the Machine Learning Department, Carnegie Mellon University, Pittsburgh, PA, USA {xiaodan1, mmv}@cs.cmu.edu

predicted sequence with the groundtruth sequence.

By integrating this novel, powerful motion GAN model with other recent visual recognition techniques, we develop a system that instruments a robot with the desired ability of predicting and demonstrating human motion. More concretely, our system takes images captured by the robot camera as input to produce the corresponding human skeleton based on real-time human pose estimation obtained with the OpenPose library [19]. Given this historical skeleton sequence, the robot then forecasts plausible motion through the motion GAN and generates a corresponding demonstration.

In summary, **our contributions** are three-fold:

- We develop a deep learning based human-robot interaction system that makes a robot capable of predicting and demonstrating human motion.
- We propose a novel motion GAN model that introduces a sequence-level discriminator and adversarial training mechanism tailored to the motion prediction task.
- We show through extensive experiments on a large-scale motion capture dataset that our motion GAN significantly outperforms state-of-the-art prediction approaches and that our entire system instruments the robot with the ability of replaying the predicted motion in a human-like manner.

II. Related Work

We briefly review the most relevant literature and discuss the differences with respect to our work.

Generative adversarial networks. GANs have shown impressive performance in image generation [20], [21], [22], video generation [23], [24], [25], and other domain tasks [26]. The key idea in GANs is an adversarial loss that forces the generator to fool the discriminator. Instead of developing new GAN objective functions as is normally the case, our goal here is to investigate how to improve human motion prediction by leveraging the GAN framework. Hence, we design a discriminator with a recurrent architecture to examine the predicted sequence from a global perspective and improve its smoothness and fidelity. Moreover, in contrast with standard GANs, our generator is the RNN encoder-decoder predictor *without any noise inputs*.

Encoder-decoder architectures. With the development of RNNs, encoder-decoder networks have been widely used in a variety of tasks, such as machine translation [27] and image caption [28]. For the human motion prediction task that we address, a 3-layer long short-term memory (LSTM-3LR) network and an encoder-recurrent-decoder (ERD) model [4] are proposed, which use curriculum learning to jointly learn a representation of pose data and temporal dynamics. High-level semantics of human dynamics are introduced into the recurrent network by modeling a human activity with a spatio-temporal graph [5]. These two approaches design their models for specific actions and restrict the training process on subsets of the motion capture dataset, such as H3.6M [29]. More recently, to explore motion prediction for general action labels, a simple residual encoder-decoder and multi-action architecture [6] is proposed by using one-hot vectors to incorporate the action label information.

However, error accumulation has been observed in the predicted sequence, since RNNs cannot recover from their own mistake [30]. This problem is alleviated by a noise scheduling scheme [31] that adds noise to the input during training [4], [5]. But this scheme makes the prediction discontinuous and makes the the hyper-parameters hard to tune. Despite their initial progress, all of these approaches only consider the prediction locally by imposing the frame-wise loss on the decoder. By contrast, we address the error accumulation problem from a sequence-level perspective by introducing a discriminator to explicitly check how human-like generated sequences are.

III. Our Approach

We now present our system that instruments a robot with the ability of predicting and demonstrating human motion, thus facilitating the human-robot interaction. As shown in Figure 2, after a person performs some action in front of the robot, the robot learns to predict and demonstrate how the person moves or acts in the near future. Our key component here is a motion GAN model, consisting of *a predictor* and *a discriminator*, that forecasts plausible and human-like motion. The predictor is an encoder-decoder network. An input sequence is passed through the encoder to infer a latent representation. This latent representation and a seed motion are then fed into the decoder to output a predicted sequence. To further evaluate the prediction fidelity from a global perspective, we introduce a discriminator that judges the realism and smoothness of the generated sequence. The predictor and the discriminator are jointly optimized in a competitive game. In the following sections, we first describe how the entire system works at the inference (deployment) stage and then discuss how we train our motion GAN.

A. Problem Formulation and Notation

Given a historical sequence, we aim to predict possible motion in the near future. The input is denoted as $\mathbf{X} = \{\boldsymbol{x_1}, \boldsymbol{x_2}, ..., \boldsymbol{x_n}\}$, where $\boldsymbol{x_i} \in \mathbb{R}^k \, (i \in [1, n])$ is a motion capture (mocap) vector at the i-th timestep that consists of a set of 3D body joint angles, n is the input sequence length, and k is the number of joint angles. Our goal is to predict the motion sequence $\widehat{\mathbf{X}} = \{\widehat{\boldsymbol{x}}_{\boldsymbol{n+1}}, \widehat{\boldsymbol{x}}_{\boldsymbol{n+2}}, ..., \widehat{\boldsymbol{x}}_{\boldsymbol{n+m}}\}$ in the next m timesteps, where $\widehat{\boldsymbol{x}}_{\boldsymbol{j}} \in \mathbb{R}^k \;\; (j \in [n+1, n+m])$ is the predicted mocap vector at the j-th timestep and m is the output sequence length. The corresponding groundtruth of the m timesteps is denoted as $\mathbf{X_{gt}} = \{\boldsymbol{x_{n+1}}, \boldsymbol{x_{n+2}}, ..., \boldsymbol{x_{n+m}}\}$.

B. Prediction and Demonstration at Inference

The first phase in our system pipeline on the robot is to capture an image from the robot. We use ROS [32] as our method of communication with the camera, and any other method of capturing an image from the robot will also work. We then send the camera image to the OpenPose library [19], which provides us with real-time pose estimations of all of the humans in the current image frame. We use an off-board

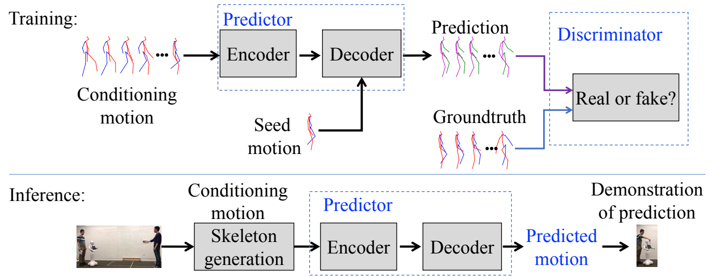

Fig. 2: An overview of our motion GAN system that teaches robots to predict human motion. Blue-red skeletons represent the input sequence and the groundtruth, and green-purple skeletons represent the prediction. During training, a conditioning sequence is fed into an *encoder* network to learn a latent representation; this latent representation and a seed motion are then fed into a *decoder* network. To further check how human-like and smooth the predicted sequence is, we design a *global discriminator* that compares the prediction with the groundtruth. Our model simultaneously optimizes the predictor and the discriminator to generate the final optimal prediction. During inference/deployment, after observing that a person performs some action in front of the camera, the robot produces the skeleton sequence, and then predicts and demonstrates how the person acts in the near future using the learned motion GAN model.

desktop computer with an Nvidia 1080 Ti GPU that allows OpenPose [19] to process images at approximately 10fps.

The next phase is to transform each human pose from 2D image coordinates into 3D points in space. There are various ways to achieve this, such as using stereo cameras to sense depth, using depth cameras, or using a model to predict the 3D positions in space. In our case, we use a depth camera that is calibrated with our RGB camera to create a point cloud. We then map the human pose coordinates in the 2D image to points in the point cloud to determine the 3D coordinates of each body part of the human skeleton.

After receiving the 3D coordinates of each body point, we transform them into the same format that was used for training on H3.6M and we then send them into our predictor.

C. Learning Motion GAN: Predictor

Human motion is modeled as sequential data and we cast the motion prediction problem as finding a mapping P from an input sequence to an output sequence. Such a sequence-to-sequence problem is typically addressed by learning an encoder-decoder network. The encoder learns a latent representation from the conditioning sequence. The decoder takes the latent representation and a seed motion as input and produces the predicted sequence.

In our motion GAN, the predictor module is responsible for learning the mapping P, so that the ℓ_2 distance between the prediction and the groundtruth is minimized:

$$\mathcal{L}_{\ell_2}\left(P\right) = \mathbb{E}\left[\left\|P\left(\mathbf{X}\right) - \mathbf{X}_{gt}\right\|_2^2\right]. \quad (1)$$

We use a similar encoder-decoder network for our predictor as in [6], given its state-of-the-art performance. Instead of working with absolute angles, the encoder takes the first order derivative velocities as input using a residual connection. A one-hot vector is introduced to indicate the action of the current input. We then concatenate the one-hot vector with the input, and feed them into the encoder. The decoder takes the output of itself as the next timestep input. The encoder and the decoder consist of gated recurrent unit (GRU) [33] cells instead of LSTM [34], since GRU is computationally more efficient. Finally, we convert the outputs of all the timesteps back to the absolute world frame, and generate the absolute angle outputs. Figure 2 shows the use of the encoder-decoder predictor in our motion GAN.

D. Learning Motion GAN: Discriminator

Previous work on human motion prediction only relies on a plain predictor. While the encoder-decoder network as the predictor can explore the temporal information of the motion in a roughly plausible way, a critical high-level fidelity examination of the prediction is missing. This leads to error accumulation and inaccurate prediction and makes the predicted motion converge to mean pose after a few frames, as observed in our experiments and previous work [5], [6]. Inspired by GANs [18], our discriminator addresses these issues through checking whether the predicted sequence is smooth and human-like from a global perspective.

A traditional GAN framework consists of two neural networks: a generative network that captures the data distribution and a discriminative network that estimates the probability of a sample being real or generated (fake). The generator is trained to generate samples to fool the discriminator and the discriminator is trained to distinguish the generation from the real samples.

Specifically, we design our discriminator $\mathcal{D}$ to distinguish between the prediction $\widehat{\mathbf{X}}$ and the groundtruth $\mathbf{X_{gt}}$. Intuitively, the discriminator evaluates how smooth and human-like the generated sequence is through directly comparing it with the groundtruth at the sequence level. Following [18], the minimax objective function is formulated as:

$$\begin{aligned}\arg\min_P \max_D \mathcal{L}_{\text{GAN}}\left(P, D\right) =& \mathbb{E}\left[\log\left(D(\mathbf{X}_{gt})\right)\right] \\ &+\mathbb{E}\left[\log\left(1 - D(P\left(\mathbf{X}\right))\right)\right].\end{aligned} \quad (2)$$

Here in an adversarial manner, P tries to minimize the objective function against D while D aims to maximize it. The quality of our motion prediction is thus judged by evaluating how well the predicted $\widehat{\mathbf{X}}$ via the predictor P fools the discriminator D.

As for the discriminator architecture, given a predicted sequence as input, we use a GRU layer to extract its latent vector representation. We then feed this vector representation into a fully-connected layer and a sigmoid layer and produce the probability whether the sequence is real or generated.

We found that it is beneficial to mix the GAN objective with the original hand-crafted ℓ_2 distance loss in Eqn. (1), which is consistent with the recent work that uses GANs for image-to-image translation [35]. Our final objective then is:

$$P^* = \arg\min_P \max_D \mathcal{L}_{\mathrm{GAN}}(P, D) + \lambda \mathcal{L}_{\ell_2}(P), \quad (3)$$

where λ is the trade-off hyper-parameter. While the objective of the discriminator remains unchanged, the predictor aims to not only fool the discriminator but also generate the prediction that is close to the groundtruth in an ℓ_2 sense.

E. Implementation Details

In our motion GAN, we use a single GRU [33] with hidden size $1{,}024$ for the encoder and decoder, respectively. Consistent with [6], we found that GRUs are computationally less-expensive and a single layer of GRU outperforms multi-layer GRUs. In addition, it is easier to train and avoids overfitting compared with the deeper models in [4], [5]. We use spatial embeddings for both the encoder and decoder. The number of GRU parameters in the discriminator is not affected by the sequence length, since sequences are fed into the GRU layer sequentially and only the embedding size (which is $1{,}024$) and the hidden size (which is $1{,}024$) affect the GRU size. Moreover, our model has the same inference time as the baseline model that only consists of a plain predictor. We use a learning rate 0.005 and a batch size 16, and we clip the gradient to a maximum ℓ_2-norm of 5. The hyper-parameter λ is cross-validated and is set as 5. We run 50 epochs. We learn our motion GAN using PyTorch [36].

IV. Experiments

In this section, we explore the use of our system to teach a Pepper robot [37] to predict and demonstrate plausible future motion when interacting with a person. To learn our motion GAN, we leverage an auxiliary, large-scale annotated mocap dataset, the Human 3.6M (H3.6M) dataset [29]. We begin with descriptions of the dataset and baselines and explain the evaluation metrics. Through extensive evaluation on H3.6M, we show that our motion GAN outperforms state-of-the-art approaches to motion prediction both quantitatively and qualitatively. Finally, we present the results on the Pepper robot, showing its ability to replay the predicted motion in a human-like, realistic manner.

Dataset. We use H3.6M [29] as an auxiliary source for training our motion GAN as well as evaluating its performance. H3.6M is an important benchmark in human motion analysis, which includes 3.6 million 3D mocap data and seven actors performing 15 varied activities, such as walking, smoking, and taking pictures. Following the experimental setup in [4], [5], [6], we downsample H3.6M by two, train on six subjects, and test on subject five. We also follow the standard split to divide the dataset into training, validation, and test sets [6]. During training, we feed our model with 50 mocap frames (2 seconds in total) and forecast the future 25 frames (1 second in total). We test on both the test set of H3.6M and the videos captured by Pepper.

A. Evaluation on the H3.6M Dataset

Table I and Figure 3 show the quantitative and qualitative comparisons between our motion GAN and state-of-the-art approaches on the test set of H3.6M, respectively.

Baselines. We compare with five recent approaches to human motion prediction based on deep RNNs: LSTM-3LR and ERD [4], SRNN [5], and sampling-based loss and residual sup. [6]. We also include a zero-velocity baseline as in [6], which constantly predicts the last observed frame. This is a simple yet strong baseline, and these learning based approaches reported that they did not consistently outperform the zero-velocity baseline.

Evaluation metrics. We evaluate the performance using the same error measurement as in [4], [5], [6] for a fair comparison, which is the Euclidean distance between the prediction and the groundtruth in the angle space. Following [16], [6], we exclude the translation and rotation of the whole body. In addition to the quantitative evaluation, we also visualize the predictions frame by frame, following a similar procedure as in [4], [5], [6].

Quantitative evaluation. Table I summarizes the comparisons between our motion GAN and the baselines on walking, eating, smoking, and discussion actions. We observe that our motion GAN significantly outperforms these deep learning based approaches, achieving the state-of-the art performance. This thus validates that the sequence-level fidelity examination of the predicted sequence is essential for more accurate motion prediction.

Moreover, Table I shows that the zero-velocity baseline performs well on complicated motions (*e.g.*, smoking and discussion) in short time periods. Although it simply uses the last observed frame as the prediction, zero-velocity is superior to the other learning based baselines, because these actions are very difficult to model. By contrast, our model consistently outperforms zero-velocity for longer time horizons ($>$ 80ms). The baseline models only verify predictions frame by frame and ignore their temporal dependencies. Our motion GAN, however, enables us to globally deal with the entire generated sequence and check how smooth and human-like it is. Such a property thus facilitates the prediction of complicated motions.

Qualitative comparisons. Figure 3 visualizes the predictions of challenging actions, including smoking and discussion, with the input motions and groundtruth motions shown in black and the generated motions shown in magenta, cyan, and blue. For reasons of space, we visualize our predictions and compare them with only the best performing baselines, sampling-based loss and residual sup. [6]. One noticeable difference between these visualizations is the degree of plausibility. The predictions of residual sup. converge to

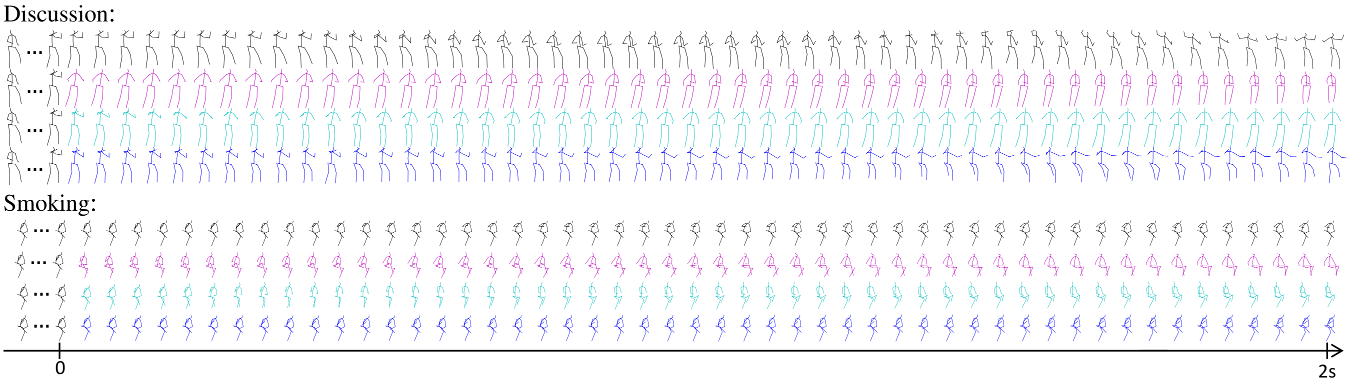

Fig. 3: Qualitative comparisons between our motion GAN and the best performing baselines, *e.g.*, sampling-based loss and residual sup. [6], for motion prediction on discussion and smoking activities. For each activity, from top to bottom: groundtruth, sampling-based loss, residual sup., and our motion GAN. For each row, the left black skeletons are the input sequences, the right black skeletons are the groundtruth, and the right colorful skeletons are the predicted sequences. Ours demonstrate more smooth and human-like predictions. **Best viewed in color with zoom.**

	Walking					Eating					Smoking					Discussion				
milliseconds	80	160	320	400	1000	80	160	320	400	1000	80	160	320	400	1000	80	160	320	400	1000
zero-velocity [6]	0.39	0.68	0.99	1.15	1.32	0.27	0.48	0.73	0.86	1.38	**0.26**	0.48	0.97	0.95	1.69	**0.31**	0.67	0.94	1.04	1.96
ERD [4]	1.30	1.56	1.84	–	2.38	1.66	1.93	2.28	–	2.41	2.34	2.74	3.73	–	3.82	2.67	2.97	3.23	–	2.92
LSTM-3LR [4]	1.18	1.50	1.67	–	2.20	1.36	1.79	2.29	–	2.82	2.05	2.34	3.10	–	3.42	2.25	2.33	2.45	–	2.93
SRNN [5]	1.08	1.34	1.60	–	2.13	1.35	1.71	2.12	–	2.58	1.90	2.30	2.90	–	3.23	1.67	2.03	2.20	–	2.43
sampling-based loss [6]	0.92	0.98	1.02	1.20	1.59	0.98	0.99	1.18	1.31	1.55	1.38	1.39	1.56	1.65	2.31	1.78	1.80	1.83	1.90	1.61
residual sup. [6]	0.28	0.49	0.72	0.81	1.03	0.23	0.39	0.62	0.76	1.08	0.33	0.61	1.05	1.15	1.50	0.31	0.68	1.01	1.09	1.69
motion GAN (Ours)	**0.27**	**0.44**	**0.63**	**0.74**	**1.00**	**0.22**	**0.35**	**0.59**	**0.70**	**1.03**	0.28	**0.48**	**0.96**	**0.94**	**1.39**	0.41	**0.63**	**0.79**	**0.91**	**1.50**

TABLE I: Detailed prediction error comparisons between our motion GAN and previously published methods, *e.g.*, zero-velocity, LSTM-3LR and ERD [4], SRNN [5], sampling-based loss and residual sup. [6] baselines, for motion prediction on walking, eating, smoking, and discussion activities of the H3.6M dataset. Our motion GAN consistently outperforms the state-of-the-art deep learning based approaches. The zero-velocity baseline achieves better performance for smoking and discussion at 80ms prediction, but our model beats zero-velocity in all the other cases, increasing well in long time horizons.

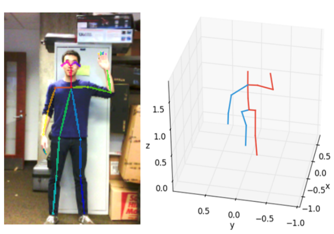

Fig. 4: OpenPose body joints from the left image are matched with a point cloud to generate our 3D skeleton output on the right.

mean poses and the predictions of sampling-based loss often drift away from the input sequences, whereas our predictions are the closest to the groundtruth. Moreover, our model performs increasingly well during the inference stage in a long-term period, which shows that our motion GAN deals well with error accumulation.

B. Motion Prediction on Pepper

We test our human motion prediction system on a real robot called Pepper from Softbank Robotics [37]. Pepper has two RGB cameras and one Asus Xtion depth sensor on its head. We first calibrate images from one RGB camera with the depth sensor to create point clouds. We then process each RGB image using OpenPose [19] to get the locations of the human joints in image coordinates, from which we map to points in the corresponding point cloud to determine the 3D skeleton points of the human in robot coordinates, as shown in Figure 4. In addition, Pepper has 6 joints on both of its arms that are fairly similar to human arms as well as 2 degrees of freedom movements in its neck [37]. We make use of all these degrees of freedom when mimicking and showing the prediction of human motion. We derive a geometric mapping from the 3D skeleton points (*i.e.*, the output of the predictor) to the angular joints on the robot, so that we can display any human motions that are within Pepper's joint limits. Figure 5 shows that Pepper successfully mimics a person's current motion and then predicts and demonstrates the person's future motion after being blinded.

V. Conclusions

In this paper, we have developed a deep learning based system that enables robots to predict and demonstrate human motion. To this end, we propose a novel motion GAN model to improve the prediction plausibility from a global perspective. A discriminator is introduced to validate the sequence-level fidelity of predicted sequences. After learning the motion GAN model from H3.6M, an auxiliary, large-scale annotated mocap dataset, we integrate it with other

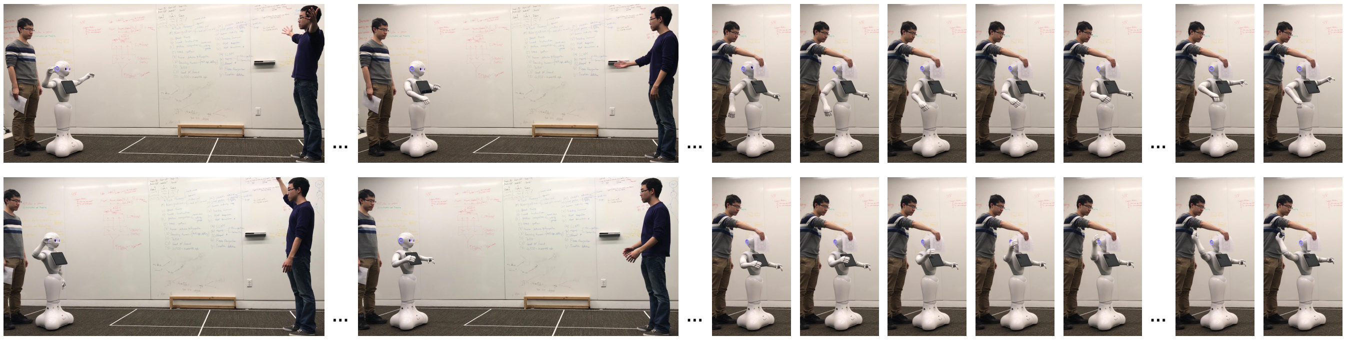

Fig. 5: Demonstrations of prediction on Pepper for discussion action (top row) and greeting action (bottom row). Pepper is mimicking the person's actions (as shown in the left two columns) until it is blinded (as shown in the third column on the left), and then begins executing motions based on its prediction (as shown in the right columns).

recent visual recognition techniques into an end-to-end prediction system. Experiments on H3.6M and a Pepper robot validate the effectiveness of our approach. In the future, we will extend our system from single subject motion to multiple subject motions and have the robot execute collaborative actions with humans by anticipating their future movements.

Acknowledgments. This research is partially sponsored by DARPA under agreements FA87501620042 and FA87501720152 and NSF grant number IIS1637927. The views and conclusions contained in this document are those of the authors only.

REFERENCES

[1] H. S. Koppula and A. Saxena, "Anticipating human activities using object affordances for reactive robotic response," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016.

[2] G. Maeda, M. Ewerton, G. Neumann, R. Lioutikov, and J. Peters, "Phase estimation for fast action recognition and trajectory generation in human–robot collaboration," *International Journal of Robotics Research*, 2017.

[3] R. Luo, R. Hayne, and D. Berenson, "Unsupervised early prediction of human reaching for human–robot collaboration in shared workspaces," *Autonomous Robots*, 2017.

[4] K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik, "Recurrent network models for human dynamics," in *IEEE International Conference on Computer Vision*, 2015.

[5] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-RNN: Deep learning on spatio-temporal graphs," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[6] J. Martinez, M. J. Black, and J. Romero, "On human motion prediction using recurrent neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[7] J. Bütepage, M. J. Black, D. Kragic, and H. Kjellström, "Deep representation learning for human motion prediction and classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[8] H. Koppula and A. Saxena, "Learning spatio-temporal structure from RGB-D videos for human activity detection and anticipation," in *International Conference on Machine Learning*, 2013.

[9] D.-A. Huang and K. M. Kitani, "Action-reaction: Forecasting the dynamics of human interaction," in *European Conference on Computer Vision*, 2014.

[10] L. Kovar, M. Gleicher, and F. Pighin, "Motion graphs," in *ACM Transactions on Graphics*, 2002.

[11] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Transactions on Intelligent Vehicles*, 2016.

[12] I. Akhter, T. Simon, S. Khan, I. Matthews, and Y. Sheikh, "Bilinear spatiotemporal basis models," *ACM Transactions on Graphics*, 2012.

[13] M. Brand and A. Hertzmann, "Style machines," in *ACM International Conference on Computer Graphics and Interactive Techniques*, 2000.

[14] J. M. Wang, D. J. Fleet, and A. Hertzmann, "Gaussian process dynamical models for human motion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008.

[15] V. Pavlovic, J. M. Rehg, and J. MacCormick, "Learning switching linear models of human motion," in *Advances in Neural Information Processing Systems*, 2001.

[16] G. W. Taylor, G. E. Hinton, and S. T. Roweis, "Modeling human motion using binary latent variables," in *Advances in Neural Information Processing Systems*, 2007.

[17] G. W. Taylor, L. Sigal, D. J. Fleet, and G. E. Hinton, "Dynamical binary latent variable models for 3D human pose tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[18] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2014.

[19] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[20] E. L. Denton, S. Chintala, and R. Fergus, "Deep generative image models using a Laplacian pyramid of adversarial networks," in *Advances in Neural Information Processing Systems*, 2015.

[21] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *International Conference on Learning Representations*, 2016.

[22] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *IEEE International Conference on Computer Vision*, 2017.

[23] X. Liang, L. Lee, W. Dai, and E. P. Xing, "Dual motion GAN for future-flow embedded video prediction," in *IEEE International Conference on Computer Vision*, 2017.

[24] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," in *Advances in Neural Information Processing Systems*, 2016.

[25] M. Mathieu, C. Couprie, and Y. LeCun, "Deep multi-scale video prediction beyond mean square error," in *International Conference on Learning Representations*, 2016.

[26] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum, "Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling," in *Advances in Neural Information Processing Systems*, 2016.

[27] R. Kiros, Y. Zhu, R. R. Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba, and S. Fidler, "Skip-thought vectors," in *Advances in Neural Information Processing Systems*, 2015.

[28] Z. Yang, Y. Yuan, Y. Wu, W. W. Cohen, and R. R. Salakhutdinov, "Review networks for caption generation," in *Advances in Neural Information Processing Systems*, 2016.

[29] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.

[30] S. Bengio, O. Vinyals, N. Jaitly, and N. Shazeer, "Scheduled sampling for sequence prediction with recurrent neural networks," in *Advances in Neural Information Processing Systems*, 2015.

[31] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *International Conference on Machine Learning*, 2009.

[32] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source robot operating system," in *IEEE International Conference on Robotics and Automation Workshop on Open Source Software*, 2009.

[33] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," in *Syntax, Semantics and Structure in Statistical Translation*, 2014.

[34] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, 1997.

[35] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[36] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," in *Advances in Neural Information Processing Systems Workshops*, 2017.

[37] M. de Jong, K. Zhang, T. Rhodes, R. Schmucker, C. Zhou, S. Ferreira, J. Cartucho, M. Veloso, and A. Roth, "Towards a robust interactive and learning social robot," in *International Conference on Autonomous Agents and Multiagent Systems*, 2018.