Teaching Robots to Predict Human Motion

Liangyan Gui[1], Kevin Zhang[2], Yu-Xiong Wang[2], Xiaodan Liang[3], José M.F. Moura[1], Manuela M. Veloso[4]

***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 has great application potential in human-robot interaction and collaboration. In this paper, we endow a robot with such ability of predicting and demonstrating human motion by leveraging recent deep learning and computer vision techniques. Our system takes images from the robot camera as input and produces the human skeleton based on real-time human pose estimation from the OpenPose library. Conditioning on this historical sequence, the robot then forecasts plausible motion through a motion predictor and generates the corresponding demonstration.**

Due to lack of high-level fidelity check, 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 novel *motion GAN predictor* achieves superior performance over the state-of-the-art deep learning approaches when evaluated on the standard Human 3.6M dataset. Based on this motion GAN predictor, 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 as shown in Figure 1. In a perfect dancing show, the robot is supposed to not only recognize but also anticipate human actions, such as accurately predicting limbs' pose and position, so that it can interact appropriately and seamlessly. The first step towards this ambitious goal is to make the robot able to *predict and demonstrate human motion* by observing human activities.

Such ability of forecasting how a human moves or acts in the near future conditioning on a series of historical movements is typically addressed in human motion prediction. In addition to human-robot interaction and collaboration [1], it has great application potential in a variety of scenarios in robotic vision, including action anticipation [2], [3], motion generation [4], and proactive decision-making in autonomous driving systems [5].

Predicting plausible human motions for diverse actions, however, is a challenging yet under-explored problem because of the uncertainty of human conscious movement and the difficulty of modeling motion dynamics. Traditional methods mainly use bilinear spatio-temporal basis models [6], Hidden Markov Models [7], Gaussian process latent variable models [8], [9], linear dynamic models [10], and Restricted Boltzmann Machines [11], [12], [13], [14]. More recently, driven by the advances of deep learning architectures and large-scale open datasets, various deep learning methods have been proposed that significantly outperform traditional approaches for predicting various actions. They formulate the task of predicting possible future motions as a sequence-to-sequence problem, which can be resolved by recurrent neural networks (RNNs) to capture the underlying temporal dependencies in the sequential data. Despite their extensive efforts on encoder-decoder type architectures (*e.g.*, encoder-recurrent-decoder (ERD) [15], [16] and residual architecture [17]), they can only predict periodic actions well (*e.g.*, walking) and show unsatisfactory performance on modeling aperiodic actions (*e.g.*, discussion) due to error accumulation.

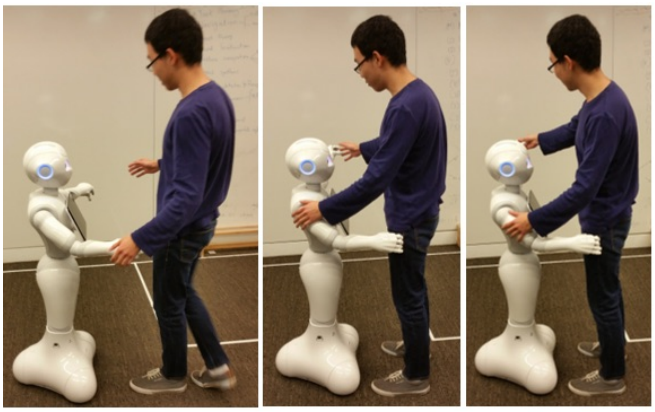

Fig. 1: Dancing with a human is a potential future use case stemming from predicting human motion.

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 human. From a higher-level viewpoint, the desired model can predict realistic periodic and aperiodic future 3D poses of a person given a series of past motions.

To achieve this, we propose a novel *motion GAN predictor 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 gen-

This material is supported by DARPA under agreement number FA8750-12-2-0291.

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

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

[3]Xiaodan Liang is with Machine Learning Department, Carnegie Mellon University, Pittsburgh, PA, USA xiaodan1@cs.cmu.edu

[4]Manuela M. Veloso is with the School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA mmv@cs.cmu.edu

Preprint submitted to 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems. Received March 1, 2018.

eration 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 training set samples and the generator tries to fool the discriminator. In the spirit of GANs, we use a residual encoder-decoder network as our generator and propose 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 predicted sequence with the groundtruth sequence.

By integrating this novel, powerful motion predictor with other recent visual recognition techniques, we develop a system that endows a robot with the desired ability of predicting and demonstrating human motion. As shown in Figue 2, our system takes images input from the robot camera and produces the human skeleton based on real-time human pose estimation from the OpenPose library [19]. Given this historical skeleton sequence, the robot then forecasts plausible motion through the motion predictor and generates the corresponding demonstration.

In summary, our contributions are three-fold:

- We develop a deep learning based system that makes a robot able to predict and demonstrate human motion.
- We propose a novel motion GAN model that introduces a sequence-level discriminator and adversarial training mechanism tailed for the motion prediction task.
- Extensive experiments on a large-scale motion capture dataset show that our motion GAN predictor significantly outperforms state-of-the-art methods and our entire system endows the robot with the ability of replaying the predicted motion in a human-like manner.

II. Related Work

We briefly review the literature that is most relevant to our task and discuss the differences.

Generative adversarial networks. GANs have shown impressive performance in image generation [20], [21], [22], video generation [23], [24], [25], and other domains [26], [27]. 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 to investigate how to improve human motion prediction by leveraging the GAN framework. Hence, we design a discriminator with RNN architectures to examine the predicted sequence from a global perspective and improve its smoothness and fidelity. Moreover, different from standard GANs, our generator is the RNN encoder-decoder *without any noise inputs*.

Encoder-decoder architecture. With the development of RNNs, encoder-decoder networks have been widely used for machine translation [28], image caption [29], and time-series prediction [30], [31], [32]. For the human motion prediction task which we address, Fragkiadaki *et al*. [32] propose a 3 layers of long short-term memory (LSTM-3LR) and an encoder-recurrent-decoder (ERD) that use curriculum learning to jointly learn a representation of pose data and temporal dynamics. Jain *et al*. [30] introduce high-level semantics of human dynamics into a recurrent network by modeling a human activity with a spatio-temporal graph. These two approaches design their models for specific actions and restrict the training process on subsets of the motion dataset, such as Human 3.6M [33]. More recently, to explore motion prediction for general action labels, Martinez *et al*. [31] develop a simple residual encoder-decoder and multi-action architecture by using one-hot vectors to incorporate the action information.

However, error accumulation has been observed in the predicted sequence, since RNNs cannot recover from their own mistake [34]. A few works [30], [32] alleviate the error accumulation problem via a noise scheduling scheme [35] by adding noise to the input during training. 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. In contrast, we address the error accumulation problem from a sequence-level perspective by introducing a discriminator to explicitly check how human-like the generated sequences are.

III. Our Approach

We now present our system that endows a robot with the ability of predicting and demonstrating human motion. As shown in Figure 2, after a person performs some action in front of the robot, the robot can learn to predict and demonstrate how the person moves or acts in the near future. Our key component here is a motion GAN predictor, consisting of *a generator* and *a discriminator*, that forecasts plausible and human-like motion. The generator 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 generated sequence as the prediction. To further evaluate the prediction fidelity from a global perspective, we introduce a novel discriminator that judges the continuation and smoothness of the generated sequence. By jointly optimizing the generator and the discriminator in a competitive game, our motion GAN predictor forecasts realistic motion, thus facilitating the human-robot interaction.

We first describe how the entire system works at the inference (deployment) stage and then discuss how we train our motion GAN predictor.

A. Problem Formulation and Notation

Given a set of historical motions, we aim to predict possible motions in the future seconds. The problem is formulated as follows. The input is a set of motions $\mathbf{X} = \{\boldsymbol{x_1}, \boldsymbol{x_2}, ..., \boldsymbol{x_n}\}$, where $\boldsymbol{x_i} \in \mathbb{R}^k \, (i \in [1, n])$ is a motion vector that consists of a set of 3D body joint angles, n is the input sequence length, and k is the number of joint angles. We ignore the global translation and rotation since only relative rotations between joints contain information of the actions. Our goal is to predict the next m timestep motion vectors $\widehat{\mathbf{X}} = \{\widehat{\boldsymbol{x}}_{\boldsymbol{n+1}}, \widehat{\boldsymbol{x}}_{\boldsymbol{n+2}}, ..., \widehat{\boldsymbol{x}}_{\boldsymbol{n+m}}\}$,

Preprint submitted to 2018 IEEE/RSJ International Conference
on Intelligent Robots and Systems. Received March 1, 2018.

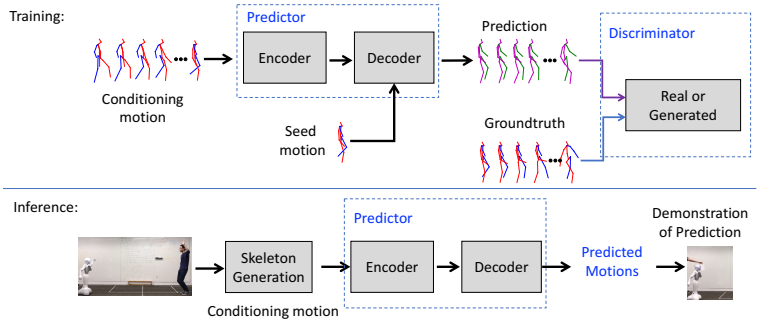

Fig. 2: An overview of our motion GAN predictor that teaches robots to predict human motion. Blue-red stick figures represent the input sequence and groundtruth, and green-purple stick figures represent predictions. During training, an input sequence is fed to an *encoder* network to encode a latent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

where $\widehat{\boldsymbol{x}}_{\boldsymbol{j}} \in \mathbb{R}^k \quad (j \in [n+1, n+m])$ is the predicted motion vector at time j and m is the output sequence length. The corresponding groundtruth 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 capturing an image from the robot. To do so, we use ROS as our method of communication with the camera, but 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 offboard desktop with an Nvidia 1080 Ti that allows OpenPose to process images at approximately 10fps.

The next phase is to transform each human's pose from 2D image coordinates into 3D points in space. There are a variety of ways to do 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 in which we use to determine the 3D coordinates of each body part of the human skeleton.

After we receive the 3D coordinates for each body point, we transform them into the same format that was used in the dataset we used for training and then we send it into our motion predictor.

C. Learning Motion GAN Predictor: Generator

Human motion is modeled as sequential data and we consider the motion prediction problem as to find a mapping P from an input sequence to an output sequence. Such sequence-to-sequence problem is typically addressed by learning an encoder-decoder network. The encoder learns a hidden representation from the input sequence. The decoder takes the hidden representation and a seed motion as inputs and produce the predicted sequence.

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

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

We use a similar encoder-decoder network for our generator as in [31], considering its state-of-the-art performance. Instead of working with absolute angles, the encoder takes the first order derivative velocities as input. A one-hot vector is introduced to indicate the action of the current input. We then concatenate the one-hot vector with the velocities, 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) [36] cells instead of LSTM [37] or other RNN variations, 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 in our motion GAN predictor.

D. Learning Motion GAN Predictor: Discriminator

Previous work on human motion prediction only relies on a plain generator. While the encoder-decoder network as generator can explore the temporal information of the motions in a roughly plausible way, a critical high-level fidelity check of the prediction is missing. This leads to error accumulation and inaccurate prediction and makes the predicted motions converge to mean position after a few frames, as observed in our experiments and previous work [31]. Inspired by generative adversarial networks (GANs) [18], we introduce a discriminator to address these issues, checking whether the generated sequence is smooth and human-like.

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. 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 sequences $\widehat{\mathbf{X}}$ and $\mathbf{X_{gt}}$. Intuitively, the discriminator evaluates how smooth and human-like the generated sequence is through directly comparing it with the groundtruth at sequence level. The objective function is formulated as:

$$\arg\min_{P}\max_{D}\mathcal{L}_{\text{GAN}}\left(P,D\right) = \mathbb{E}\left[\log\left(D(\mathbf{X})\right)\right] + \mathbb{E}\left[\log\left(1-D(P\left(\mathbf{X}\right))\right)\right]. \quad (2)$$

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

The discriminator architecture is as follows. Given a predicted sequence as input, we use an encoder to extract its vector representation. We then feed the 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 [38]. Our final objective then is:

$$P^{*} = \arg\min_{P}\max_{D}\mathcal{L}_{\text{GAN}}\left(P,D\right) + \lambda\mathcal{L}_{\ell_2}\left(P\right), \quad (3)$$

where λ is the trade-off parameter. While the discriminator's job remains unchanged, the generator aims to not only fool the discriminator but also to be close to the groundtruth prediction in an ℓ_2 sense.

E. Implementation Details

In our motion GAN predictor, we use a single gated recurrent unit (GRU) [36] with hidden size 1024 for the encoder and decoder, respectively. 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 deeper models as used in [30], [32]. We use spatial embedding for both the encoder and decoder. Despite an additional discriminator, the number of GRU parameters in the discriminator is not affected by the sequence length, since sequences are fed into GRU cells sequentially and only the embedding size (which is 1024) and the hidden size (which is 1024) affect the GRU size. Hence, our model has the same inference time as the baseline model that only consists of a plain generator.

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. λ is cross-validated. We run 50 epochs. We learn our motion GAN predictor using PyTorch [39], which takes 35ms for forward processing and back-propagation per iteration on an NVIDIA GPU.

IV. Experiments

In this section, we explore the use of our system to teach a pepper robot [40] to predict and demonstrate human future motion when interacting with a person. To learn our motion GAN predictor, we leverage an auxiliary, large-scale annotated motion capture (mocap) dataset, the Human 3.6M dataset [33]. We begin with descriptions of the dataset and baselines and explain the evaluation metrics. Through extensive evaluation on Human 3.6M, we show that our motion GAN predictor outperforms the state-of-the-art deep learning approaches for motion prediction both quantitatively and qualitatively. Finally, we provide the results on the pepper robot, showing human-like, realistic motion prediction and demonstration.

Dataset. We use the Human 3.6M dataset [33] as an auxiliary source for training our motion GAN predictor as well as evaluating its performance with the state-of-the-art approaches in motion prediction. Human 3.6M is a large-scale, publicly available dataset including 3.6 million 3D motion capture data and it is an important benchmark in human motion analysis. Human 3.6M consists of seven actors performing 15 varied activities, such as walking, smoking, engaging in a discussion, and taking pictures. We follow the experimental setup of [32]. Specifically, we downsample Human 3.6M by two, train on six subjects, and test on subject five. We split the dataset to three parts following [32]: a training set, a validation set, and a test set. During training and validation, we feed our model with 50 mocap frames (2 seconds in total) and forecast the future 25 frames (1 second in total). We test both on the test set of Human 3.6M and the videos captured by Pepper.

A. Evaluation on Human 3.6M Dataset

Table I and Figure 3 show the quantitative and qualitative comparisons of our motion GAN with the state-of-the-art approaches on the test set of Human 3.6M, respectively.

Baselines. We compare with five recent approaches for human motion prediction based on deep RNNs: *LSTM-3LR* and *ERD* by Fragkiadaki *et al*. [32], *SRNN* by Jain *et al*. [30], and *Sampling-based loss* and *Residual sup.* by Martinez *et al*. [31]. Similar as [31], we also consider a *zero-velocity* baseline that constantly predicts the last observed frame. This is a simple but strong baseline, and all of the learning based approaches reported that their models did not consistently outperform the zero-velocity baseline.

Evaluation metrics. We evaluate the performance using the same error measurement as in [30], [32], [31] for a fair comparison, which is the Euclidean distance between the predicted motions and the groundtruth motions in the angle space. We exclude the translation and rotation of the whole body since this information is independent of the actions themselves. In addition to the quantitative evaluation, we also visualize the predictions frame by frame, following a similar procedure as in [30], [32], [31].

Quantitative evaluations. 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 the deep learning approaches, achieving state-of-the art performance.

Preprint submitted to 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems. Received March 1, 2018.

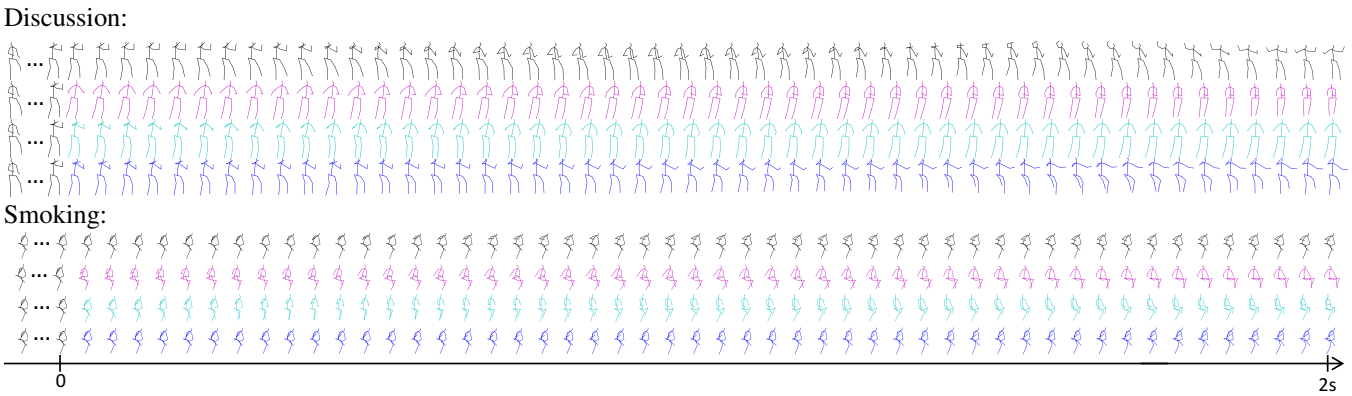

Fig. 3: Qualitative comparisons between our motion GAN and the best-performing baselines, *e.g.*, *Sampling-based loss* and *Residual sup.* [31], for motion prediction of discussion and smoking activities. For each activity, from top to bottom: groundtruth, *Sampling-based loss*, *Residual sup.*, and 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 prediction. (Best viewed in color with zoom.)

This thus validates that the sequence-level fidelity examination of the predicted sequence from a global perspective is essential for precise 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 constantly predicts the last observed frame, *zero-velocity* is superior to the other learning based baselines, because these actions are very difficult to model. In contrast, our model consistently outperforms *zero-velocity* for longer time horizon ($>$ 80ms). The baseline models only verify predictions one frame by one frame separately and ignore their temporal dependencies. Our motion GAN predictor enables us to deal with the entire generated sequence globally and check how smooth and human-like it is. Such property especially facilities the prediction of complicated motions.

Qualitative comparisons. Figure 3 visualizes the prediction of the 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 with the best-performing baselines, *Sampling-based loss* and *Residual sup.* One noticeable difference between these visualizations is the degree of jump (*i.e.*, discontinuity) between the seed frame and the first few predicted frames. The jump in *Sampling-based loss* and *Residual sup.* are severe, whereas the jump in our prediction is relatively small. Moreover, our model performs increasingly well during the inference step in long-term period, which shows that our motion GAN can deal 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 [40]. Pepper has two RGB cameras and one Asus Xtion depth sensor on its head. We use one of the front RGB cameras along with the depth camera to generate the 3D skeleton points of the humans using OpenPose [19], which is 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 movement in its neck [40]. We make use of all these degrees of freedom when mimicking and showing the predictions 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 we can display any human motions that are within Pepper's joint limits. Figure 5 shows that Pepper successfully mimics a human's current motion and then predicts and demonstrates the human's future motion after being blinded.

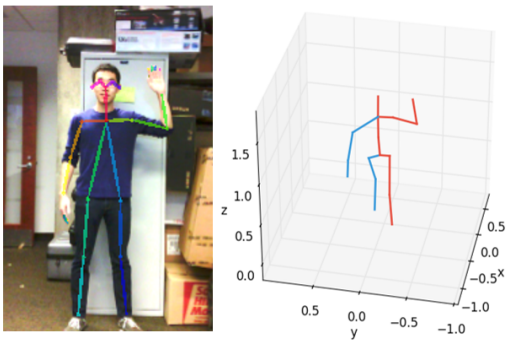

Fig. 4: OpenPose body keypoints from the left image are matched with a point cloud to generate our 3D skeleton output on the right.

V. Conclusions

In this paper, we develop a deep learning based system that enables robots to predict and demonstrate human motion. We present a novel motion GAN predictor model to improve the plausibility of predicted motion sequences from a global perspective. A discriminator is proposed to model the sequence-level fidelity of predicted sequences. After learning the motion GAN predictor using Human 3.6M, an auxiliary, large-scale annotated dataset, we integrate it with other recent visual recognition techniques to develop an end-to-end prediction system. Experiments on the Human

	Walking					Eating					Smoking					Discussion				
millisecond	80	160	320	400	1000	80	160	320	400	1000	80	160	320	400	1000	80	160	320	400	1000
Zero-velocity	0.39	0.68	0.99	1.15	1.33	0.27	0.48	0.73	0.86	1.27	**0.26**	0.48	0.97	0.95	1.41	**0.31**	0.67	0.94	1.04	1.81
ERD [32]	0.93	1.18	1.59	1.78	1.92	1.27	1.45	1.66	1.80	1.87	1.66	1.95	2.35	2.42	2.87	2.27	2.47	2.68	2.76	2.73
LSTM-3LR [32]	0.77	1.00	1.29	1.47	1.77	0.89	1.09	1.35	1.46	1.65	1.34	1.65	2.04	2.16	2.56	1.22	1.49	1.83	1.93	2.79
SRNN [31]	0.81	0.94	1.16	1.30	1.69	0.97	1.14	1.35	1.46	1.91	1.45	1.68	1.94	2.08	2.01	1.22	1.49	1.83	1.93	2.26
Sampling-based loss [31]	0.92	0.98	1.02	1.20	1.40	0.98	0.99	1.18	1.31	1.59	1.38	1.39	1.56	1.65	1.77	1.78	1.80	1.83	1.90	2.01
Residual sup. [31]	0.27	0.47	0.70	0.78	1.12	0.23	0.39	0.62	0.76	1.30	0.33	0.61	1.05	1.15	1.92	0.31	0.68	1.01	1.09	1.81
motion GAN (Ours)	**0.27**	**0.41**	**0.63**	**0.74**	**0.91**	**0.22**	**0.35**	**0.59**	**0.70**	**1.22**	0.28	**0.48**	**0.96**	**0.94**	**1.19**	0.41	**0.63**	**0.79**	**0.91**	**1.71**

TABLE I: Detailed prediction error comparisons between our motion GAN and previously published methods, *e.g.*, *zero-velocity*, *LSTM-3LR* and *ERD* [32], *SRNN* [30], *Sampling-based loss* and *Residual sup.* [31] baselines, for motion prediction of walking, eating, smoking, and discussion activities on the Human3.6M dataset. Our motion GAN consistently outperforms the state-of-the-art deep learning 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.

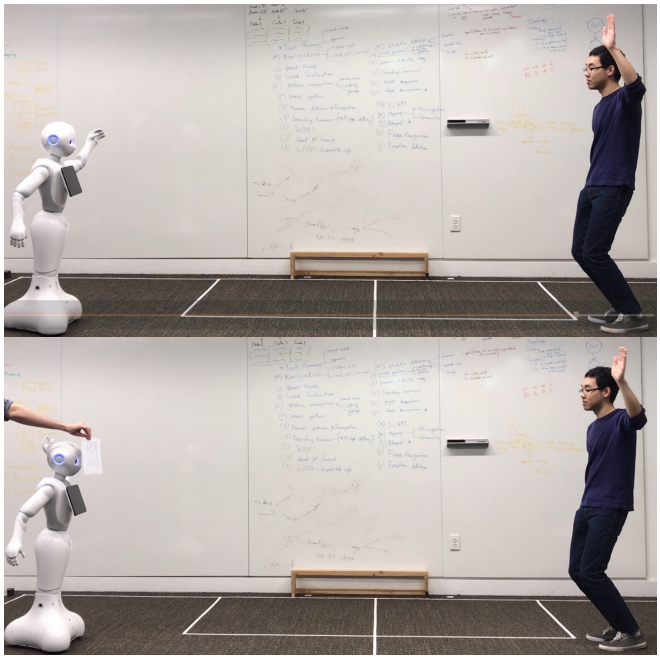

Fig. 5: Pepper is mimicking the human on the right in the top image until it is blinded and then begins executing motions based on its prediction of the human's motions.

3.6M dataset and a pepper robot validate the effectiveness of our approach. In the future, we will extend the proposed framework from single subject motion to multiple-subject motion and have the robot execute collaborative actions with the human by anticipating their future movements.

REFERENCES

[1] H. S. Koppula and A. Saxena, "Anticipating human activities using object affordances for reactive robotic response," *TPAMI*, 2016.

[2] H. Koppula and A. Saxena, "Learning spatio-temporal structure from RGB-D videos for human activity detection and anticipation," in *ICML*, 2013.

[3] D.-A. Huang and K. M. Kitani, "Action-reaction: Forecasting the dynamics of human interaction," in *ECCV*, 2014.

[4] L. Kovar, M. Gleicher, and F. Pighin, "Motion graphs," in *ACM (TOG)*, 2002.

[5] 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," *TIV*, 2016.

[6] I. Akhter, T. Simon, S. Khan, I. Matthews, and Y. Sheikh, "Bilinear spatiotemporal basis models," *ACM (TOG)*, 2012.

[7] M. Brand and A. Hertzmann, "Style machines," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, 2000.

[8] J. M. Wang, D. J. Fleet, and A. Hertzmann, "Gaussian process dynamical models for human motion," *TPAMI*, 2008.

[9] R. Urtasun, D. J. Fleet, A. Geiger, J. Popović, T. J. Darrell, and N. D. Lawrence, "Topologically-constrained latent variable models," in *ICML*, 2008.

[10] V. Pavlovic, J. M. Rehg, and J. MacCormick, "Learning switching linear models of human motion," in *NIPS*, 2001.

[11] G. W. Taylor, G. E. Hinton, and S. T. Roweis, "Modeling human motion using binary latent variables," in *NIPS*, 2007.

[12] G. W. Taylor and G. E. Hinton, "Factored conditional restricted Boltzmann machines for modeling motion style," in *ICML*, 2009.

[13] I. Sutskever, G. E. Hinton, and G. W. Taylor, "The recurrent temporal restricted Boltzmann machine," in *NIPS*, 2009.

[14] G. W. Taylor, L. Sigal, D. J. Fleet, and G. E. Hinton, "Dynamical binary latent variable models for 3D human pose tracking," in *CVPR*, 2010.

[15] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural computation*, 1989.

[16] P. Rodriguez, J. Wiles, and J. L. Elman, "A recurrent neural network that learns to count," *Connection Science*, 1999.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

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

[19] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *CVPR*, 2017.

[20] E. L. Denton, S. Chintala, R. Fergus, *et al.*, "Deep generative image models using a Laplacian pyramid of adversarial networks," in *NIPS*, 2015.

[21] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

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

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

[24] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," in *NIPS*, 2016.

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

[26] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," in *ICML*, 2016.

[27] 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 *NIPS*, 2016.

[28] R. Kiros, Y. Zhu, R. R. Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba, and S. Fidler, "Skip-thought vectors," in *NIPS*, 2015.

[29] Z. Yang, Y. Yuan, Y. Wu, W. W. Cohen, and R. R. Salakhutdinov, "Review networks for caption generation," in *NIPS*, 2016.

[30] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-RNN: Deep learning on spatio-temporal graphs," in *CVPR*, 2016.

[31] J. Martinez, M. J. Black, and J. Romero, "On human motion prediction using recurrent neural networks," in *CVPR*, 2017.

[32] K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik, "Recurrent network models for human dynamics," in *ICCV*, 2015.

[33] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human 3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments," *TPAMI*, 2014.

[34] S. Bengio, O. Vinyals, N. Jaitly, and N. Shazeer, "Scheduled sampling for sequence prediction with recurrent neural networks," in *NIPS*, 2015.

[35] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *ICML*, 2009.

[36] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

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

[38] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *CVPR*, 2017.

[39] 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," *NIPS Workshops*, 2017.

[40] S. Robotics. (2017) NAOqi API and Pepper documentation. [Online]. Available: http://doc.aldebaran.com/2-5