# Generating Animated Videos of Human Activities from Natural Language Descriptions

**Angela S. Lin**[1]*, **Lemeng Wu**[1]*, **Rodolfo Corona**[2],
**Kevin Tai**[1], **Qixing Huang**[1], **Raymond J. Mooney**[1]
[1] Department of Computer Science, The University of Texas at Austin, Austin, TX 78712, USA
[2] University of Amsterdam, Amsterdam, Netherlands
alin@cs.utexas.edu, lm.wu@utexas.edu, r.coronarodriguez@uva.nl
kevin.r.tai@utexas.edu, huangqx@cs.utexas.edu, mooney@cs.utexas.edu

## Abstract

Generating realistic character animations is of great importance in computer graphics and related domains. Existing approaches for this application involve a significant amount of human interaction. In this paper, we introduce a system that maps a natural language description to an animation of a humanoid skeleton. Our system is a sequence-to-sequence model that is pretrained with an autoencoder objective and then trained end-to-end.

## 1 Introduction

Developing automatic tools to generate visual content is a fundamental problem in computer graphics. The primary way artists currently create CGI animation sequences is by specifying a series of key frames for the characters. A key frame is a character pose at a particular point in time. This is a time-consuming and tedious process since each key frame is specified by manually moving the character into the desired position. This paper presents a method for automatically mapping a natural language (NL) description of a human activity to an animated video, specifically a sequence of 3D human skeletal poses that can be used to animate a CGI character. This allows the generation of animations with minimal user effort. Human-provided NL descriptions of human activities for which motion-capture (mocap) data is available [29] is used to train the system.

The problem of mapping between language and other modalities such as images and videos has attracted significant recent attention. In particular, there has been considerable work on deep neural networks for mapping videos to NL descriptions [25, 40, 45]; however, there is very little work on the inverse problem of text-to-video. The small amount of work on generating images and videos from text [21, 26, 33, 34, 47] generates pixel-level images that are sometimes low-quality, rather than concise 3D models that can be flexibly rendered into a variety of high-definition visual content using standard computer graphics techniques. The small amount of work on generating 3D graphics models from text [5–8, 36] focuses on static scenes rather than animations. There have been a few prior projects on mapping natural language descriptions of human activities to motion sequences using mocap data [30, 44], however, they do not specifically focus on generating animated videos for graphics applications and do not evaluate the quality of the generated animations using human judges.

Text-to-video is inherently harder than video-to-text since it requires generating long, real-valued, high-dimensional sequences from short, discrete ones, instead of the other way around. Also, language is inherently ambiguous, so there may be many animations that fit the same description. In addition, the amount of labeled mocap data is limited compared to images and videos. Mocap

*Note: (*) denotes co-first authorship

32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

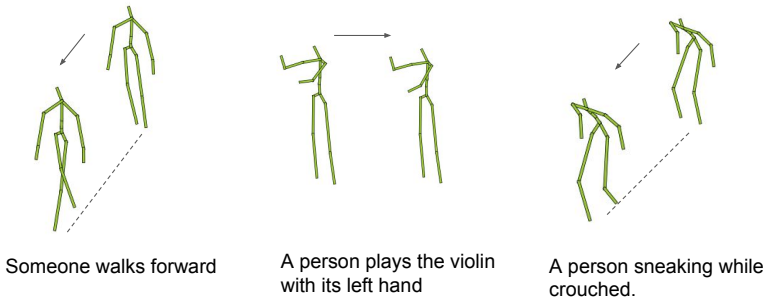


Figure 1: Examples of gold-standard description-animation pairs. The dotted lines show movement through space and the arrows show movement through time.

systems are expensive to setup and existing datasets do not use the same joint markers on the actors, making the resulting animations incompatible with each other. Finally, there is a large imbalance in the number of videos for different types of action in the dataset – there are many more videos of walking compared to dancing. Examples from the dataset are shown in Figure 1.

Due to these complexities, our initial attempts to use standard recurrent neural network (RNN) sequence-to-sequence (seq2seq) models [37] used for video-to-text met with limited success. To address these problems, we use a neural autoencoder to learn a compact representation of human motions by training on mocap data without NL descriptions, and then use an RNN to map descriptions into this motion representation.

The remainder of the paper describes the details of our method and presents a detailed evaluation of the animations generated for held-out test examples by comparing them to gold-standard animations, and by asking crowd-sourced human judges to evaluate their faithfulness to the given descriptions.

## 2 Related Work

**Video Captioning** There has been a growing amount of recent work on generating NL descriptions of videos. Venugopalan et al. [40] developed a seq2seq model for video captioning using convolutional neural nets (CNNs) to encode the frames and RNNs to map the sequence of frames to a sequence of words. Subsequent works [2, 25, 41] have improved this model using language models, pretrained word embeddings, attention, and hierarchical modelling.

**Animation Synthesis from Text** There has been some recent work on generating animations from text, however they use a data representation [39] where the hip position of the character has a fixed 3D location, therefore the character moves in place rather than along the floor. Plappert et al. [30] propose a seq2seq model for mapping text to a series of Gaussian distributions representing the joint angles of the character. Yamada et al. [44] use an autoencoder for text and an autoencoder for animations with a shared latent space to generate animations from text. These models are incomplete as the descriptions of the motions often include information about how the character moves in the global coordinate frame as well as how the character moves in the local coordinate frame.

**Direct Animation Synthesis** Since collecting motion capture data is expensive, generating new animation sequences from existing data has a long history in the computer graphics community [18, 31, 43]. Recently, researchers have used deep learning techniques to tackle this problem. One line of work has been on synthesizing animations under user constraints such as foot placement locations and times [13–15]. There have also been several works on synthesizing the continuation of an animation sequence given a few frames at the beginning of the animation [12, 22, 23]. In contrast to our problem setting, the input and output domain are the same in these methods, therefore their networks only need to model the uncertainty and wide range of futures, rather than also modeling the gap between the text and animation domains.

**Motion Controllers** In addition to methods that directly synthesize animations, there have been methods that learn policies for controlling simulated characters. The motivation for this approach is that animated characters should obey the physics of their world, thus the training procedure should include the complexity of the environment. One classical approach for generating animations is by

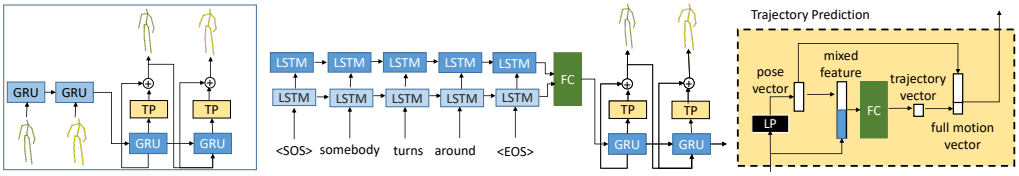


Figure 2: Our network architecture. (Left) Autoencoder architecture for the animation. (Middle) Network architecture for our full pipeline. (Right) Network architecture for the trajectory prediction module. LP indicates the linear projection layer and FC indicates the fully connected layer.

modeling the body using physics models [20, 48]. Another approach to this problem is to learn a policy for executing different actions using reinforcement learning. Some of these methods learn policies from scratch [38, 46] while others try to mimic the behavior in examples [3, 24, 27, 28, 42].

## 3 Approach

In the same spirit as other seq2seq modeling tasks (e.g., machine translation [32] and video captioning [40]), we design an end-to-end neural network $f_\theta$, where $\theta$ encodes the network parameters, by combining a module for encoding the input NL description, a module for decoding the result into the output skeleton animation, and an intermediate module for connecting these two modules (See Figure 2). The encoder is a standard two layer stacked LSTM [11] and the decoder architecture is based on the GRU [9] with residual connections proposed by Martinez et al. [23]. We use the data representation proposed by Holden et al. [14], which factors out the skeleton animation as the combination of the character's pose as represented by the joint positions of the character with respect to the local coordinate frame and the character's trajectory of movement in the global coordinate frame. Drawing inspiration from the network architecture proposed by Agrawal et al. [1], we separated the pose prediction and trajectory prediction so that the trajectory prediction is conditioned on the pose prediction in addition to the GRU output.

**Network training.** Since the animation representation is higher dimensional compared to the NL descriptions, we first pretrain the decoder with an autoencoder loss. This is similar to the pretraining step for the task of machine translation. For the autoencoder pretraining step, we use a combination of the KIT Motion-Language Dataset [29] and the Human3.6M dataset [4, 16]. We then use the paired NL-mocap data from the KIT Motion-Language Dataset [29] and additional paired data that we collected on Amazon Mechanical Turk (AMT) using a video segmentation and annotation tool designed for dense video event captioning [19] to train the entire network. The loss function for both training steps is the L2 distance between the pose and trajectory of the gold-standard animation and predicted animation. We train the model until convergence using Adam [17].

## 4 Experimental Results

This section presents experiments that use both an automatic metric and crowd-sourced human judgments to evaluate our generated animations and compare them to several baselines.

**Baseline Methods**

**1. Nearest Neighbor (NN)** Our simplest baseline is a standard TF-IDF bag-of-words nearest neighbor method. First, we vectorize all sentences using TfidfVectorizer in Scikit-learn (*scikit-learn.org*). Then, to generate a video for a test description, we find the closest vectorized description in the training set using cosine similarity and return the corresponding animation.

**2. Plappert et al. [30] (P)** This is one of the aforementioned algorithms that also maps NL descriptions to mocap sequences. We modified their code to incorporate our new data while keeping everything else intact and trained their model using the hyperparameter settings listed in their paper.

**Evaluation Metrics**

**Dynamic Time Warped Mean Absolute Error (DTW-MAE)** We compare an animation generated for a description to the gold-standard (GS) animation and compute the mean absolute error. To compare animations of different lengths, we use a dynamic time warping algorithm [35] to stretch

| | DTW-MAE | DTW-MAE-T |
|---|---|---|
| NN | $9.80 \pm 5.79$ | $9.76 \pm 5.77$ |
| P | N/A | $8.44 \pm 3.99$ |
| Ours | $9.74 \pm 4.34$ | $9.71 \pm 4.32$ |

| M1 vs. M2 | M1 Win Rate | M2 Win Rate |
|---|---|---|
| P vs. GS | 0.105 | 0.895 |
| Ours vs. GS | 0.196 | 0.804 |
| Ours vs. P | 0.790 | 0.210 |

Table 1: Results on the test set. For DTW-MAE, lower is better. For the win rate, higher is better.

the shorter animation to the length of the longer animation. We then compute the mean absolute error between all of the corresponding poses and trajectories, averaging across animation frames. We computed this metric on all 805 gold-standard description-animation pairs in the test set. Plappert et al. [30]'s method only considers the pose of the animation, therefore we also present the DTW-MAE without the trajectory information (DTW-MAE-T) on the nearest neighbors baseline and our method for a fairer comparison.

**Human Evaluation** Since many different animations can be a good depiction of the described activity, similarity in the joint position space may not correlate with the overall quality of the generated result. Therefore, we also conduct a crowd-sourced human evaluation of the generated animations using AMT. We evaluate the generated animations for faithfulness, analogous to machine translation evaluations of fidelity. Our Human Intelligence Task (HIT) presents AMT workers with two videos for the same description, randomly chosen from: gold-standard, our method, and Plappert et al.'s method, along with the description. The AMT worker is then asked to select the animation that is a better depiction of the activity described in the text. The win rate is defined as the number of comparisons won by the method (M) divided by the total number of comparisons for a particular pair of methods.

In each HIT, we have workers rate three pairs of videos. For quality control, one of the pairs is a "verification" test to determine if the worker is paying attention. We generate a pool of 20 verification tasks from the validation set videos by randomly pairing a gold-standard description-animation pair with a gold-standard animation from a different pair. We manually check that the selected distractor animation does not depict the described activity. For each HIT, we include a randomly selected verification pair and discard data from the HIT if it is answered incorrectly. We selected a subset of 200 gold-standard description-animation pairs from the test set for the human evaluation experiment. 17 percent of the evaluations were thrown out due to failure on the verification task.

**Discussion** As we can see from Table 1, our method outperforms Plappert et al. [30]'s method in the human evaluation study. This may not be a fair comparison for Plappert et al. [30]'s method as many descriptions describe movement in the global coordinate frame and cannot be acted out while standing in place, e.g., "A person walking in a circle to the left." The gold-standard animations greatly outperforms both models, showing that there is still more room for improvement for automatic methods. In a preliminary human evaluation study, we found that the nearest neighbor baseline is surprisingly strong for this task. This may be due to the fact that there are different actors performing the same activity in the dataset. We do not use the activity information when determining the dataset split, therefore animations of the same activity may be in both the training set and the test set. The main failure cases of our model are producing animations that fail to depict the description for less well-represented activities or producing animations that are physically impossible, e.g., the human figure glides along the floor instead of taking distinct steps.

The automatic evaluation metric, DTW-MAE, does not agree well with human judgment of animation quality. This demonstrates the need for human evaluation on graphics tasks and better automatic metrics that capture semantic meaning [10]. Martinez et al. [23] found that predicting the average pose at every time step is a strong baseline when comparing animations using mean absolute error of joint angles. Upon visual inspection of results produced by Plappert et al. [30], we found that the human figure is static in many of the animations, which may cause the mean absolute error to be low.

## 5 Conclusions and Future Work

We present an end-to-end sequence-to-sequence model for generating human motion animations from natural language descriptions. In the future, we plan to improve our model by improving our loss function to capture more semantic meaning and explore physically-based controller approaches to generate more realistic animations.

## References


[1] Pulkit Agrawal, Ashvin V Nair, Pieter Abbeel, Jitendra Malik, and Sergey Levine. Learning to Poke by Poking: Experiential Learning of Intuitive Physics. In *Advances in Neural Information Processing Systems (NIPS)*, pages 5074–5082, 2016.

[2] Nicolas Ballas, Li Yao, Chris Pal, and Aaron Courville. Delving Deeper into Convolutional Networks for Learning Video Representations. In *Proceedings of ICLR*, 2016.

[3] Glen Berseth, Cheng Xie, Paul Cernek, and Michiel Van de Panne. Progressive Reinforcement Learning with Distillation for Multi-Skilled Motion Control. In *Proceedings of ICLR*, 2018.

[4] Cristian Sminchisescu Catalin Ionescu, Fuxin Li. Latent Structured Models for Human Pose Estimation. In *Proceedings of the International Conference on Computer Vision*, 2011.

[5] Angel Chang, Manolis Savva, and Christopher Manning. Semantic parsing for text to 3d scene generation. In *Proceedings of the ACL 2014 Workshop on Semantic Parsing*, pages 17–21, 2014.

[6] Angel Chang, Manolis Savva, and Christopher D Manning. Learning spatial knowledge for text to 3d scene generation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2028–2038, 2014.

[7] Angel Chang, Will Monroe, Manolis Savva, Christopher Potts, and Christopher D Manning. Text to 3D Scene Generation with Rich Lexical Grounding. In *Proceedings of ACL*, 2015.

[8] Kevin Chen, Christopher B Choy, Manolis Savva, Angel X Chang, Thomas Funkhouser, and Silvio Savarese. Text2Shape: Generating Shapes from Natural Language by Learning Joint Embeddings. *arXiv preprint arXiv:1803.08495*, 2018.

[9] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. In *Proceedings of SSST-8*, pages 103–111, 2014.

[10] Huseyin Coskun, David Joseph Tan, Sailesh Conjeti, Nassir Navab, and Federico Tombari. Human Motion Analysis with Deep Metric Learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[11] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to Forget: Continual Prediction with LSTM. 1999.

[12] Partha Ghosh, Jie Song, Emre Aksan, and Otmar Hilliges. Learning Human Motion Models for Long-Term Predictions. In *Proceedings of the 2017 International Conference on 3D Vision (3DV)*, pages 458–466. IEEE, 2017.

[13] Ikhsanul Habibie, Daniel Holden, Jonathan Schwarz, Joe Yearsley, and Taku Komura. A Recurrent Variational Autoencoder for Human Motion Synthesis. *Proceedings of BMVC*, 2017.

[14] Daniel Holden, Jun Saito, and Taku Komura. A Deep Learning Framework for Character Motion Synthesis and Editing. *ACM Trans. Graph.*, 35(4):138:1–138:11, July 2016. ISSN 0730-0301. doi: 10.1145/2897824.2925975. URL `http://doi.acm.org/10.1145/2897824.2925975`.

[15] Daniel Holden, Taku Komura, and Jun Saito. Phase-Functioned Neural Networks for Character Control. *ACM Transactions on Graphics (TOG)*, 36(4):42, 2017.

[16] 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 Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, 2014.

[17] Diederik P Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[18] Lucas Kovar, Michael Gleicher, and Frédéric Pighin. Motion Graphs. In *ACM SIGGRAPH 2008 classes*, page 51. ACM, 2008.

[19] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 706–715, 2017.

[20] Taesoo Kwon and Jessica K Hodgins. Momentum-Mapped Inverted Pendulum Models for Controlling Dynamic Human Motions. *ACM Transactions on Graphics (TOG)*, 36(1):10, 2017.

[21] Yitong Li, Martin Renqiang Min, Dinghan Shen, David Carlson, and Lawrence Carin. Video Generation from Text. In *Proceedings of AAAI-2018*, 2018.

[22] Zimo Li, Yi Zhou, Shuangjiu Xiao, Chong He, Zeng Huang, and Hao Li. Auto-Conditioned Recurrent Networks for Extended Complex Human Motion Synthesis. *Proceedings of ICLR*, 2018.

[23] Julieta Martinez, Michael J Black, and Javier Romero. On Human Motion Prediction Using Recurrent Neural Networks. In *Proceedings of CVPR*, pages 4674–4683. IEEE, 2017.

[24] Josh Merel, Yuval Tassa, Sriram Srinivasan, Jay Lemmon, Ziyu Wang, Greg Wayne, and Nicolas Heess. Learning Human Behaviors from Motion Capture by Adversarial Imitation. *arXiv preprint arXiv:1707.02201*, 2017.

[25] Pingbo Pan, Zhongwen Xu, Yi Yang, Fei Wu, and Yueting Zhuang. Hierarchical Recurrent Neural Encoder for Video Representation with Application to Captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1029–1038, 2016.

[26] Yingwei Pan, Zhaofan Qiu, Ting Yao, Houqiang Li, and Tao Mei. To Create What You Tell: Generating Videos from Captions. In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 1789–1798. ACM, 2017.

[27] Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel van de Panne. DeepMimic: Example-Guided Deep Reinforcement Learning of Physics-Based Character Skills. *ACM Transactions on Graphics (TOG)*, 37(4), 2018.

[28] Xue Bin Peng, Angjoo Kanazawa, Jitendra Malik, Pieter Abbeel, and Sergey Levine. SFV: Reinforcement Learning of Physical Skills from Videos. *ACM Trans. Graph.*, 37(6), November 2018.

[29] Matthias Plappert, Christian Mandery, and Tamim Asfour. The KIT Motion-Language Dataset. *Big Data*, 4(4):236–252, December 2016. doi: 10.1089/big.2016.0028. URL `http://dx.doi.org/10.1089/big.2016.0028`.

[30] Matthias Plappert, Christian Mandery, and Tamim Asfour. Learning a Bidirectional Mapping Between Human Whole-Body Motion and Natural Language using Deep Recurrent Neural Networks. *Robotics and Autonomous Systems*, 109:13–26, 2018.

[31] Katherine Pullen and Christoph Bregler. Motion Capture Assisted Animation: Texturing and Synthesis. *ACM Transactions on Graphics (TOG)*, 21:501–508, 2002.

[32] Prajit Ramachandran, Peter J. Liu, and Quoc V. Le. Unsupervised Pretraining for Sequence to Sequence learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 383–391, 2017.

[33] Tiago Ramalho, Tomáš Kociskỳ, Frederic Besse, SM Eslami, Gábor Melis, Fabio Viola, Phil Blunsom, and Karl Moritz Hermann. Encoding Spatial Relations from Natural Language. *arXiv preprint arXiv:1807.01670*, 2018.

[34] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative Adversarial Text to Image Synthesis. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, pages 1060–1069, 2016.

[35] Stan Salvador and Philip Chan. FastDTW: Toward Accurate Dynamic Time Warping in Linear Time and Space. *Intelligent Data Analysis*, 11(5):561–580, 2007.

[36] Stephan Streuber, M Alejandra Quiros-Ramirez, Matthew Q Hill, Carina A Hahn, Silvia Zuffi, Alice O'Toole, and Michael J Black. Body Talk: Crowdshaping Realistic 3D Avatars with Words. *ACM Transactions on Graphics (TOG)*, 35(4):54, 2016.

[37] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to Sequence Learning with Neural Networks. In *Advances in Neural Information Processing Systems*, pages 3104–3112, 2014.

[38] Russ Tedrake, Teresa Weirui Zhang, and H Sebastian Seung. Stochastic Policy Gradient Reinforcement Learning on a Simple 3D Biped. In *Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004)*, volume 3, pages 2849–2854. IEEE, 2004.

[39] Ömer Terlemez, Stefan Ulbrich, Christian Mandery, Martin Do, Nikolaus Vahrenkamp, and Tamim Asfour. Master Motor Map (MMM)—Framework and Toolkit for Capturing, Representing, and Reproducing Human Motion on Humanoid Robots. In *2014 14th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 894–901. IEEE, 2014.

[40] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to Sequence–Video to Text. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4534–4542, 2015.

[41] Subhashini Venugopalan, Lisa Anne Hendricks, Raymond Mooney, and Kate Saenko. Improving LSTM-based Video Description with Linguistic Knowledge Mined from Text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2016.

[42] Ziyu Wang, Josh S Merel, Scott E Reed, Nando de Freitas, Gregory Wayne, and Nicolas Heess. Robust imitation of diverse behaviors. In *Advances in Neural Information Processing Systems (NIPS)*, pages 5320–5329, 2017.

[43] Douglas J Wiley and James K Hahn. Interpolation Synthesis for Articulated Figure Motion. In *Proceedings of the Virtual Reality Annual International Symposium*, pages 156–160. IEEE, 1997.

[44] Tatsuro Yamada, Hiroyuki Matsunaga, and Tetsuya Ogata. Paired Recurrent Autoencoders for Bidirectional Translation Between Robot Actions and Linguistic Descriptions. *IEEE Robotics and Automation Letters*, 3(4):3441–3448, 2018.

[45] Haonan Yu, Jiang Wang, Zhiheng Huang, Yi Yang, and Wei Xu. Video Paragraph Captioning using Hierarchical Recurrent Neural Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4584–4593, 2016.

[46] Wenhao Yu, Greg Turk, and C Karen Liu. Learning Symmetric and Low-Energy Locomotion. *ACM Transactions on Graphics (TOG)*, 37(4):144, 2018.

[47] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaolei Huang, Xiaogang Wang, and Dimitris Metaxas. StackGAN: Text to Photo-Realistic Image Synthesis with Stacked Generative Adversarial Networks. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 5907–5915, 2017.

[48] Victor Brian Zordan and Jessica K Hodgins. Motion Capture-Driven Simulations That Hit and React. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 89–96. ACM, 2002.