A Motion Matching-based Framework for Controllable Gesture Synthesis from Speech

Ikhsanul Habibie
Mohamed Elgharib
Kripashindu Sarkar
ihabibie@mpi-inf.mpg.de
elgharib@mpi-inf.mpg.de
ksarkar@mpi-inf.mpg.de
MPI Informatics
Germany

Ahsan Abdullah
Simbarashe Nyatsanga
Michael Neff
aabdullah@ucdavis.edu
simnyatsanga@ucdavis.edu
mpneff@ucdavis.edu
UC Davis
USA

Christian Theobalt
theobalt@mpi-inf.mpg.de
MPI Informatics
Germany

ABSTRACT

Recent deep learning-based approaches have shown promising results for synthesizing plausible 3D human gestures from speech input. However, these approaches typically offer limited freedom to incorporate user control. Furthermore, training such models in a supervised manner often does not capture the multi-modal nature of the data, particularly because the same audio input can produce different gesture outputs. To address these problems, we present an approach for generating controllable 3D gestures that combines the advantage of database matching and deep generative modeling. Our method predicts 3D body motion by sequentially searching for the most plausible audio-gesture clips from a database using a k-Nearest Neighbors (k-NN) algorithm that considers the similarity to both the input audio and the previous body pose information. To further improve the synthesis quality, we propose a conditional Generative Adversarial Network (cGAN) model to provide a data-driven refinement to the k-NN result by comparing its plausibility against the ground truth audio-gesture pairs. Our novel approach enables direct and more varied control manipulation that is not possible with prior learning-based counterparts. Our experiments show that our proposed approach outperforms recent models on control-based synthesis tasks using high-level signals such as motion statistics while enabling flexible and effective user control for lower-level signals. [1]

CCS CONCEPTS

• **Computing methodologies → Motion processing**.

KEYWORDS

gesture synthesis, character control, audio-driven pose estimation

ACM Reference Format:
Ikhsanul Habibie, Mohamed Elgharib, Kripashindu Sarkar, Ahsan Abdullah, Simbarashe Nyatsanga, Michael Neff, and Christian Theobalt. 2022. A Motion Matching-based Framework for Controllable Gesture Synthesis from Speech. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Proceedings (SIGGRAPH '22 Conference Proceedings), August 7–11, 2022, Vancouver, BC, Canada.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3528233.3530750

1 INTRODUCTION

Creating human-like 3D avatars is important in order to provide immersive experiences in virtual worlds. Advances in 3D vision and graphics can now synthesize human-like virtual characters that emulate various aspects of human anatomy, thus potentially simplifying the production of personalized avatars. Designing an easy and accessible way to control such avatars can improve social interactivity between users and such avatars in shared virtual environments.

An appealing approach for developing intuitive character control is to synthesize character gestures from input speech. However, developing an algorithm for speech to gesture synthesis is known to be a challenging task [Alexanderson et al. 2020; Ferstl et al. 2019; Ginosar et al. 2019; Habibie et al. 2021]. This is partly due to the nature of the audio-to-gesture relationship, where many different gesture sequences may be appropriate for a given speech input. Hence, training a regression-based model in a supervised manner can lead to unnatural "averaged" gesture results as the consequence of regressing multiple outcomes of a single input signal. While recent methods [Ferstl et al. 2019; Ginosar et al. 2019; Habibie et al. 2021] use adversarial learning to mitigate the problem of "averaged" synthesis, they provide very limited options for controlling the output, and hence they predict only one particular motion sequence for every speech input. Since human gesture is known to be related to the personality and internal state of the speaker [Smith and Neff 2017], the ability to control body motion based on a specific input signal such as their emotional state can significantly improve the usability of the method. Recently, generative models were employed to introduce probabilistic synthesis to allow some degree of high-level gesture control [Alexanderson et al. 2020]. However, such methods typically need high quality training data to work well, are slow to train, and require separate pre-trained models to produce different types of control, thus limiting their usability when multiple aspects of the control signal should be varied.

In this work, we propose a new approach for controllable 3D body gesture generation from speech inspired by the popularMotion

[1] Project webpage: http://vcai.mpi-inf.mpg.de/projects/SpeechGestureMatching/

SIGGRAPH '22 Conference Proceedings, August 7–11, 2022, Vancouver, BC, Canada
© 2022 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-9337-9/22/08.
https://doi.org/10.1145/3528233.3530750

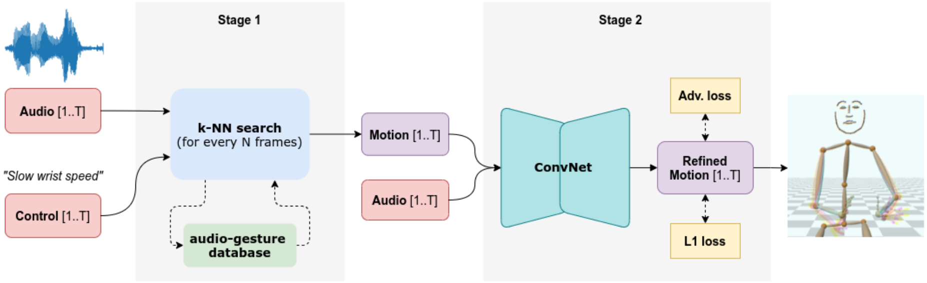


Figure 1: Our proposed pipeline consists of two main stages. In Stage 1, we first employ a k-Nearest Neighbor search to find the most plausible sequence considering the audio and previous pose similarity in the database. At any given time step, additional information can be provided to incorporate further control over of the synthesis output. The 3D gesture generated through Stage 1 is then passed to a conditional GAN trained to produce a refined gesture sequence by comparing the output against real audio-gesture sequences.

Matching algorithm commonly found in locomotion synthesis [Büttner and Clavet. 2015] (see Figure 1). At the core of our method is a novel k-Nearest Neighbor-based (k-NN) algorithm that selects short clips from a database and is specifically designed to leverage the similarity both in the audio and in the motion space to ensure a continuous, natural, and synchronous gesture output. We further improve the quality of the initial synthesis by passing the k-NN output into a conditional Generative Adversarial Network (cGAN). The cGAN is conditioned on both the audio and the motion synthesized by the k-NN as input and is tasked with producing a new motion that looks similar to the real ground truth motion in the database. This allows the network to perform correction on any less plausible motion generated by the k-NN, especially around the transition boundary between segments. Our novel gesture synthesis formulation can be naturally extended to select motion based on control information by only considering motion candidates that match the control criteria. Unlike the most related approach of Alexanderson et al. [2020], we can control our synthesis at any particular time and with various types of conditioning without the need to re-train our model for every given type of control signal. Furthermore, our control extends beyond high-level signals such as motion statistics, to include direct per-frame manipulation which can be exploited for various interesting applications such as semantic-level control.

Experiments show that our proposed model clearly outperforms related control-based audio-driven synthesis [Alexanderson et al. 2020], both in terms of naturalness and audio synchronization. Furthermore, even in the absence of control, our technique achieved better synthesis quality than the previous state-of-the-art approach [Habibie et al. 2021]. In summary, our contributions are three-fold: *1)* We propose a novel Motion Matching-based algorithm for gesture synthesis from speech, *2)* A deep generative modeling approach to resynchronize and enhance the synthesis quality from the k-NN by leveraging the whole training data that cannot be fully exploited using database features alone, *3)* We significantly outperform previous control-based gesture synthesis method [Alexanderson et al. 2020] while using a more interpretable design that enables greater set of control signals than other previous learning-based approaches, thus facilitating a wider range of potential applications.

2 RELATED WORK

Human gestures are known to be highly correlated to speech and often convey meaningful information. Here we will briefly discuss various techniques that have been proposed to learn the correlation between gesture and speech. Like many other data-driven models [Ferstl et al. 2019; Ginosar et al. 2019; Habibie et al. 2021], our main goal is to model the generation of beat gestures, which are the repetitive motion used to emphasize certain parts of the speech [McNeill 2000]. However, our formulation also allows us to generate a specific type of gesture at a particular time by leveraging additional information to produce body motion beyond beat gestures.

2.1 Gesture Synthesis from Speech

The literature of audio-driven gesture synthesis can mostly be grouped into either rule-based [Cassell 2000; Cassell et al. 1994, 2001; Lee and Marsella 2006] or data-driven [Alexanderson et al. 2020; Chiu and Marsella 2011; Ginosar et al. 2019; Hasegawa et al. 2018; Levine et al. 2010, 2009] approaches. In this section, we will focus on data-driven techniques as they are more relevant to our work. Early gesture synthesis methods leveraged stochastic models such as Dynamic Bayesian Networks (DBN) to learn from data. Levine et al. [2009] used a Hidden Markov Model to generate gestures by iteratively selecting the most plausible motion sub-sequence from a set of clusters. This work was extended to employ reinforcement learning and Conditional Random Fields [Levine et al. 2010] to improve synthesis quality. Other work demonstrated that DBNs can also be used to model eyebrow and head generation from speech [Mariooryad and Busso 2012] and incorporate discourse functions [Sadoughi and Busso 2019]. Other stochastic models such as RBM [Chiu and Marsella 2011] and GPLVM [Chiu and Marsella 2014] have also shown to be a feasible framework for speech-gesture synthesis.

Compared to the aforementioned classical data-driven models, deep learning approaches have shown to be more effective at learning from a large amount of data. Hasegawa et al. [2018] incorporated an LSTM-based neural network to design a speech-driven model that predicts a sequence of gestures. An LSTM-based model was used by Shlizerman et al. [2018] to translate audio features of musical instruments such as piano and violin into a sequence of 2D body keypoints. Yoon et al. [2019] used a variant of LSTM models to predict 2D gestures from text transcripts of speech curated from "in-the-wild" TED videos tracked using a 2D monocular pose estimator. Similarly, Ginosar et al. [2019] used OpenPose [Cao et al. 2017] to track 2D body and hand pose annotations from 140 hours of speech videos and learned an adversarial-based convolutional model to predict the gesture. Habibie et al. [2021] extended this work by using 3D annotations on the same dataset while also incorporating 3D facial expression. Furthermore, they also updated the adversarial loss by incorporating the synchronization with the audio. Ferstl et al. [2019] proposed using a gesture phase classifier and multi-objective adversarial losses to improve gesture generation quality. Ferstl et al. [2020] used an LSTM network to analyse the predictability of several gesture parameters from speech, such as velocity, acceleration, and arm swivel. Leveraging this model, Ferstl et al. [2021] proposed a gesture synthesis approach by selecting gesture clips from a database that have the best matching gesture parameters when compared to the corresponding parameters predicted from the speech input. Lee et al. [2019] introduced a large motion capture dataset of the full body and hands of two people in a face-to-face, spontaneous conversation. Ahuja et al. [2020] proposed a deep mixture model to simultaneously learn gesture content and speaker style from multiple speakers. Work by Kucherenko et al. [2020] and Yoon et al. [2020] demonstrated the capability of learning-based gesture synthesis models to incorporate text information. Bhattacharya et al. [2021] explored the use of GAN to perform co-speech gesture synthesis by analyzing affective cues. Li et al. [2021] proposed a variational autoencoder based approach to enable stochastic gesture synthesis from speech input. Yoon et al. [2021] proposed a framework to control certain aspects of the speech gesture synthesis by actively involving human in-the-loop.

Most of the aforementioned deep learning methods are deterministic in nature, and their generation are not straightforward to control. The most relevant work that can achieve both probabilistic and control-aware functionalities is the MoGlow approach by Alexanderson et al. [2020]. Their approach was based on a generative model known as normalizing flows. This allowed their method to generate multiple plausible gestures by sampling from a latent space, and the input can be conditioned on a signal to learn high-level controls from data, e.g. hand height and hand velocity. However, training such models is typically slow and requires high quality data, making it challenging to train on large scale but noisy "in-the-wild" data captured from monocular video. Compared to MoGlow, our proposed method can provide more stable, controlled gesture synthesis without resorting to a complex, learning-based model. Our method can also adapt to different types of control signals on the fly. For example, at test time, our method can generate fast gestures in the first half of the sequence and low-hand gestures in the second half, while MoGlow requires two separate models to produce gesture with different control types.

2.2 Motion Synthesis and Control

Since our work utilizes Motion Matching, which is commonly used in video games, here we survey methods for motion synthesis and control. Graph-based algorithms were amongst the most popular choice for classical data-driven character animation and control [Arikan and Forsyth 2002; Kovar et al. 2002; Lee et al. 2002; Safonova and Hodgins 2007]. For example, a motion graph can be constructed to model connections and transitions between motion clips in a dataset, and motion synthesis can be achieved by traversing the graph. However, graph-based approaches do not scale well with data and at times can be imprecise and unresponsive to the control input. Instead of directly using the actual pose representation in the search space of the graph-based approaches, Lee et al. [2014] proposed *motion fields* to generalize the motion representation into a higher dimensional vector space, enabling more responsive synthesis. To achieve control, a reinforcement learning (RL) model was trained to find the best action that can satisfy user's input. Büttner and Clavet. [2015] proposed Motion Matching to further simplify this process and approximate the RL algorithm by casting it as a k-Nearest Neighbor search. Motion generation and control are performed by selecting the most suitable clip from the database that best matches the previous pose and user's desired trajectory. Holden et al. [2020] proposed a fully learning-based approach to Motion Matching for locomotion by emulating the database look-up process with a neural network regressor. This is possible for locomotion control since there is a clear mapping between motion trajectory and the corresponding 3D joint position of the character. Unfortunately, this approach does not work well for speech-to-gesture synthesis. This is because there are multiple plausible 3D gestures that can be mapped from a single speech input, making it very difficult to replace database search using a standard regression algorithm. In our work, we used a classical nearest neighbor-based approach to perform synthesis. Since the k-NN approach makes direct use of the input data, our algorithm can be extended to allow various types of gesture control by restricting access to subsets of this data. A learning-based model is further used to refine and re-synchronize the outcome of the k-NN.

3 PROPOSED METHOD

Our system consists of two main components. In this section, we will first describe the design of our nearest neighbor (k-NN) algorithm for gesture synthesis and control. We then describe our design choices to improve the k-NN result through the use of a cGAN to transform the gesture into a more natural and synchronized motion.

3.1 Nearest Neighbor-based Gesture Synthesis

Our k-NN is inspired by the Motion Matching algorithm which has become a popular method of choice for locomotion synthesis in the gaming industry due to its flexibility and good visual quality [Büttner and Clavet. 2015; Holden et al. 2020]. Direct selection over the database using k-NN naturally avoids the problem of regression to the mean, while also providing more flexible control options. Multi-modal synthesis can be generated by either selecting different k-values or choosing different pose initializations.

3.1.1 Input/Output Parameters.

Our k-NN algorithm takes as input a sequence of audio features $\mathbf{F} = [\mathbf{f}_0, \mathbf{f}_1, ..., \mathbf{f}_{T-1}]$, one frame of initial previous pose features $\mathbf{p}_{-1}$, and optionally a sequence of control masks $\mathbf{C} = [\mathbf{c}_0, \mathbf{c}_1, ..., \mathbf{c}_{T-1}]$, where T is the number of frames in the sequence. The output is a sequence of 3D body poses $\mathbf{G} = [\mathbf{g}_0, \mathbf{g}_1, ..., \mathbf{g}_{T-1}]$. Each audio feature frame $\mathbf{f}_t$ and pose feature frame $\mathbf{p}_t$ encode information about the relevant future frames. The audio feature consists of the first 13 coefficients of the Mel-frequency cepstral coefficients (MFCC) as well as the audio log mean energy. The pose feature is derived from the 3D locations of the wrists, elbows, index finger roots, and little finger roots of both hands.

To find the output sequence, the input features need to be matched with the sequences in the *Matching Database*. The database is constructed from a collection of ground truth audio and 3D body motion pairs. It consists of M sequences of training audio features $\mathcal{F} = [\tilde{\mathbf{F}}^0, ..., \tilde{\mathbf{F}}^{M-1}]$, training pose features $\mathcal{P} = [\tilde{\mathbf{P}}^0, ..., \tilde{\mathbf{P}}^{M-1}]$, as well as the corresponding gesture sequence $\mathcal{G} = [\tilde{\mathbf{G}}^0, ..., \tilde{\mathbf{G}}^{M-1}]$ where $\tilde{\mathbf{F}}^m = [\tilde{\mathbf{f}}_0^m, ..., \tilde{\mathbf{f}}_{T_{match}-1}^m]$, $\tilde{\mathbf{P}}^m = [\tilde{\mathbf{p}}_0^m, ..., \tilde{\mathbf{p}}_{T_{match}-1}^m]$, and $\tilde{\mathbf{G}}^m$ $= [\tilde{\mathbf{g}}_0^m, ..., \tilde{\mathbf{g}}_{T_{match}-1}^m]$. The sequences in the database are prepared by segmenting the original videos into $T_{match} = 64$ frame chunks.

3.1.2 *Proposed Search Algorithm.*

To find the optimal output gesture sequence $\mathbf{G}$ from the database, we consider both the similarity with respect to the current test audio features $\mathbf{f}_t$ as well as the previously searched pose features $\mathbf{p}_{t-1}$ for every N frame interval. Please note that each feature frame contains information of future frames. In the first iteration, the previous pose feature $\mathbf{p}_{-1}$ is initialized by either randomly sampling a frame from the database or set to be the mean pose. Weighting the importance of audio and pose terms is a challenging task since their quantities cannot be directly compared. Our algorithm resolves this issue by aggregating the similarity rank of the candidates in both audio and pose space. For every iteration, a gesture sequence candidate is picked if the sum of its audio and pose similarity rank is the lowest compared to other candidates.

To speed up search computation, we pre-select one best candidate from each training sequence $(\tilde{\mathbf{F}}^m, \tilde{\mathbf{P}}^m)$ in the database based on either the pose similarity ("pose pre-selected") or audio similarity ("audio pre-selected") before scoring them based on both pose and audio similarity scores. Here, we will only describe the "pose pre-selected" k-NN version of the algorithm, which we also use as input to the later stage, even though we also find the result of "audio pre-selected" compelling.

The gesture selection is performed at a regular interval of $N = 8$ frames. During each iteration, given the current frame t, we first pre-select M pose sequence candidates $\{\hat{\mathbf{p}}^0_{0:(N-1)}, \hat{\mathbf{p}}^1_{0:(N-1)}, ..., \hat{\mathbf{p}}^{M-1}_{0:(N-1)}\}$ from the database by comparing the Euclidean distance to the previous pose feature $\mathbf{p}_{t-1}$ at frame $t-1$. We also measure the similarity of the audio features by comparing the current test audio feature $\mathbf{f}_t$ against the corresponding audio feature frame candidates from the database $\{\hat{\mathbf{f}}^0_{0:(N-1)}, \hat{\mathbf{f}}^1_{0:(N-1)}, ..., \hat{\mathbf{f}}^{M-1}_{0:(N-1)}\}$ using a cosine distance metric.

To aggregate both metrics, we first create two separate rankings based on audio match quality and pose match quality using their similarity scores. Afterward, we combine both the pose similarity and audio similarity ranks for every candidate by adding their respective ranks in both lists. This combined rank list $R_{combined}$ is then used as the new metric to select the best gesture sequence. A gesture output candidate $\mathbf{g}^*_{0:(N-1)}$ is selected as the best output gesture $\mathbf{g}_{t:(t+N-1)}$ for the current frame t if its corresponding audio and pose features result in the lowest rank in $R_{combined}$. Algorithm 1 in the supplementary material summarizes our approach.

3.1.3 *Enabling Gesture Control with k-NN search.*

Since the algorithm performs explicit comparison between features in the database, we can naturally extend this process to enforce high to mid-level control over the synthesis, allowing a more direct and interpretable way to achieve a desired behavior. For example, simulating gesture generation that follows a certain motion statistic can be achieved by simply labeling parts of the training data that satisfy the criteria. For example, to produce a sequence of body gestures where the left hand is always higher than the specified threshold $\mathbf{r}$, the search can be restricted to only consider frames where hand heights are higher than $\mathbf{r}$. Formally, this controlled synthesis can be performed by using a binary control mask matrix $\mathbf{c} \in \{0,1\}^{M \times T_{match}}$ which is constructed by checking if the gesture at a particular frame t is eligible according to the control signal or not. This allows us to effectively limit the search space to the desired data. In practice, we only check the first and the last frame of each search window ($N = 8$ frames long) to allow a wider range of possible options. Unconstrained gesture synthesis can be seen as a special case where the value of $\mathbf{c}$ is always 1 at every frame. Compared to the controlled synthesis performed by MoGlow [Alexanderson et al. 2020], our control design is more flexible as we can mix different control criteria at either the same or different frames seamlessly without requiring hours of re-training the neural network for each given criteria. A range of masks C can easily be calculated for various control features of interest. Please refer to the supplementary material for the pseudo-code of our proposed k-NN algorithm.

3.2 Gesture Resynchronization using cGAN

Our experiments suggest that the 3D gesture produced in the first stage appear natural and in-sync with the audio. However, since the similarity metric of the k-NN serves as an approximation to the real audio-gesture correspondence, its predicted frames may not always lead to the most optimal solution. Furthermore, the use of window-based search at a regular interval may also limit the ability of the algorithm to consider longer correlations. To address these issues, we enhance the synthesis quality by passing the output of the first stage into a learned conditional Generative Adversarial Network (cGAN). Adversarial-based generative models are known for their ability to produce high quality synthesis that closely matches the real data distribution, especially if they are also guided by a conditional input signal [Isola et al. 2017; Mirza and Osindero 2014]. To this end, we train a generator network G which transforms an audio-gesture pair $(\bar{\mathbf{F}}, \mathbf{G}_{kNN})$ generated by the k-NN into another pair $(\bar{\mathbf{F}}, \mathbf{G}_{syn})$ which has similar characteristics to the real audio-gesture pairs $\{(\bar{\mathbf{F}}^m, \mathbf{G}^m_{real})\}_{m=0}^{M-1}$ in the training data. We denote every feature $\bar{\mathbf{f}}_t \in \mathbb{R}^{28}$ in a sequence $\bar{\mathbf{F}}$ as the concatenation of the MFCC feature $\mathbf{m}_t \in \mathbb{R}^{14}$ with its first derivative. A separate discriminator network D is trained to classify between the real audio-gesture sequence pairs from the real 3D gesture distribution and the fake audio-gesture pairs generated by the k-NN. Both networks are trained in an alternating fashion to compete with each other. Once the training converges, the generator is expected to produce more realistic 3D body and hand gestures given the conditioning 3D gesture input from the k-NN. As the task of the generator is to update the initial 3D gesture produced by the k-NN, we found that using parent-relative representation for $\mathbf{G}_{real}$, $\mathbf{G}_{kNN}$, and $\mathbf{G}_{sync}$ leads to a more stable result. We used the Wasserstein GAN loss formulation [Arjovsky et al. 2017] with gradient penalty [Gulrajani et al. 2017]:

$$\mathcal{L}_{Adv}(G,D) = \mathbb{E}_{\bar{\mathbf{F}},\mathbf{G}_{real}}\left[D(\bar{\mathbf{F}},\mathbf{G}_{real})\right] - \mathbb{E}_{\bar{\mathbf{F}},\mathbf{G}_{syn}}\left[D(\bar{\mathbf{F}},\mathbf{G}_{syn})\right], \quad (1)$$

where $\mathbf{G}_{syn} = G(\bar{\mathbf{F}},\mathbf{G}_{kNN})$. Furthermore, the gradient penalty is defined as follows:

$$\mathcal{L}_{GP}(G,D) = \mathbb{E}_{\mathbf{G}_{syn}}\left[(\parallel \nabla_{\mathbf{G}_{syn}} D(\mathbf{G}_{syn}) \parallel -1)^2\right]. \quad (2)$$

To ensure that the output of the generator can be guided by the 3D gesture produced by the k-NN, we also use a reconstruction loss $\mathcal{L}_{Rec} = \mathcal{L}_1(\mathbf{G}_{kNN},\mathbf{G}_{syn})$ to encourage gesture similarity between the k-NN and generator output.

$$\mathcal{L} = w_1 \cdot \mathcal{L}_{Rec} + w_2 \cdot \mathcal{L}_{Adv}(G,D) + w_3 \cdot \mathcal{L}_{GP}(G,D). \quad (3)$$

The architecture of our network closely follows the fully convolutional design used by Habibie et al. [2021] due to its flexibility in dealing with temporal data of arbitrary length. The generator takes as input MFCC features $\bar{\mathbf{F}} \in \mathbb{R}^{B\times C_m \times T_{match}}$ and the 3D gesture $\mathbf{G}_{kNN} \in \mathbb{R}^{B\times C_g \times T_{match}}$, where B is the batch size, while C_m and C_g are respectively the size of the audio and gesture features. For the generator G, the encoder of the network is comprised of 8 blocks of 1D convolution, 1D batch normalization (BN) [Ioffe and Szegedy 2015], and ReLU activation functions [Nair and Hinton 2010]. A Max Pooling layer is used after every second block with the exception of the last. The decoder consist of 8 blocks mirroring the encoder, each of which contains [1D conv, 1D BN, ReLU] layers except for the last one which uses just a single 1D convolution to produce the final resynchronized gesture $\mathbf{G}_{syn}$. The decoder blocks are interleaved with an upsampling layer after every second block. On the other hand, the discriminator takes as input the MFCC features $\bar{\mathbf{F}} \in \mathbb{R}^{B\times C_m \times T_{match}}$ and either the real $\mathbf{G}_{real} \in \mathbb{R}^{B\times C_g \times T_{match}}$ or generated $\mathbf{G}_{kNN} \in \mathbb{R}^{B\times C_g \times T_{match}}$ gesture sequence (see Equation 1). The discriminator D consists of 6 blocks of 1D convolution, 1D instance normalization, and a leaky ReLU activation function with an Average Pooling layer after every second block, followed by a fully-connected layer at the end to produce a scalar value which rates the similarity of the input with respect to the real audio-gesture distribution. Please see our supplementary material for more detail regarding the architecture.

3.3 Training Details

We used the 3D annotated version [Habibie et al. 2021] of the speech-to-gesture data curated by Ginosar et al. [2019]. The *Matching Database* consists of 9624 unique gesture $\tilde{\mathbf{G}}$, audio $\tilde{\mathbf{F}}$, and pose $\tilde{\mathbf{P}}$ feature sequences, each of which is $T_{match} = 64$ frames, as is commonly used for this dataset. This is equal to more than 11 hours of data. To train the cGAN resynchronization network, we prepared a new dataset which contains audio-gesture sequences with a strided overlap of 5 frames between each consecutive samples. Since the adversarial loss compares the real 3D gesture sequence $\mathbf{G}_{real}$ with the "fake" or k-NN-generated 3D gesture sequence $\mathbf{G}_{kNN}$, we also need to generate the fake gesture "ground truth" $\mathbf{G}_{kNN}$. To achieve this, we run the k-NN algorithm over the training sequences to generate $\mathbf{G}_{kNN}$ by using the training audio features as input. To ensure that the network can handle different gesture characteristics from different k-nearest neighbors, we sampled 50% of our data from $k = 1$ while the rest 50% are uniformly from $k = 2$ to $k = 15$. In all experiments, the initial pose feature for the k-NN is generated by randomly sampling a feature frame from the database. The network is trained over 300,000 iterations using Adam optimizer [Kingma and Ba 2015] with a learning rate of $1e-4$. The hyperparameters w_1, w_2 and w_3 are set to be 0.1, 1, and 100, respectively.

4 EXPERIMENT AND EVALUATION

To verify the feasibility of our proposed approach, we evaluate its performance with various control signals. We discuss the versatility of our design in achieving various types of control without the need for re-training, and show how our method can be extended to perform semantically meaningful gesture synthesis. Finally, in the absence of a control signal, we also show that our method achieves better performance compared to the prior state-of-the-art approach [Habibie et al. 2021].

Since multiple motions may be correct for a given audio sequence, there are no well established metrics for assessing performance. Hence, we resort to user studies for performance evaluation which is the most standard evaluation protocol [Alexanderson et al. 2020; Habibie et al. 2021; Kucherenko et al. 2021; Yoon et al. 2019].

Because gesture style is known to be speaker-specific, most prior works train and test their models on the same speaker. To this end, we use a single speaker (John Oliver) of the 3D annotated version [Habibie et al. 2021] of the in-the-wild Berkeley speech-gesture dataset [Ginosar et al. 2019] to train and test the examined methods. In our case, using a single subject also makes it easier for the participants to recognize their speaking style.

All user study participants were recruited from Amazon Mechanical Turk. Before the study, each user was shown two real video examples of the speaker along with the 3D face, body, and hand tracking results. The users were asked to ignore the synthesis of the facial expression, which always use 3D tracked ground truth keypoints. During each study, the 3D rendered gesture videos, along with their audio, were shown one-by-one to the user. The video playback control was disabled once the user clicked the play button, and the user was not able to proceed until the playback has been completed. At the end of every video, each user was asked to rate the quality of the gesture synthesis using a seven point scale, ranging from 1 (lowest) to 7 (highest). The users were asked to rate each video based on two prompts: *1)* Does the clip appear natural and the gesture follow the speaking style of the speaker?, and *2)* Are the gesture and the audio well synchronized? We ran multiple preliminary tests to ensure that the objective of the study is well understood by the participants based on their feedback. More details regarding the user study instruction are shown in the supplementary material. All comparison videos are 24 seconds long and are uniquely and randomly sampled for each user from the original test dataset of Ginosar et al. [2019].

4.1 Evaluation of High-level Gesture Control

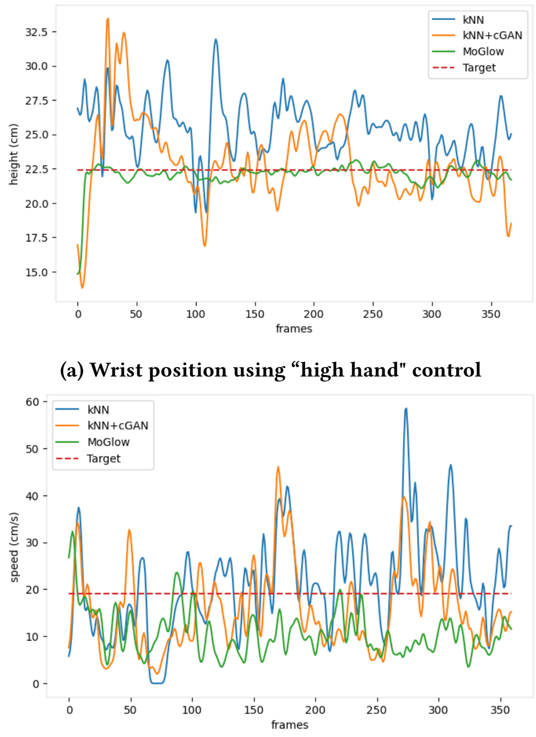
4.1.1 Subjective Evaluation. We evaluate and compare the performance of our proposed k-NN+cGAN method against the state-of-the-art, audio-driven, control-based gesture synthesis approach of MoGlow [Alexanderson et al. 2020]. Here, we examine three

Table 1: A user study for evaluating various control-based synthesis techniques. Our proposed approach was consistently rated as more natural and more in-sync than MoGlow [Alexanderson et al. 2020].

Method	Naturalness ↑	Synchrony. ↑
GT	5.95 ± 1.06	6.00 ± 1.17
Ours Height	**5.25 ± 1.26**	**5.10 ± 1.53**
MoGlow Height	4.79 ± 1.45	4.71 ± 1.65
Ours Speed	**5.33 ± 1.36**	**5.25 ± 1.55**
MoGlow Speed	5.20 ± 1.35	5.21 ± 1.36
Ours Symmetry	**5.21 ± 1.16**	**5.33 ± 1.12**
MoGlow Symmetry	4.77 ± 1.58	4.70 ± 1.62

different control signals: left wrist height, left wrist speed, and wrist height symmetry. To this end, we re-train the MoGlow model on the 3D annotated version Habibie et al. [2021] of the Berkeley speech-to-gesture data Ginosar et al. [2019] based on their publicly available code. We performed Gaussian smoothing on the training data to ease the training process and used the best qualitative result after conducting grid search over around 30 different parameter combinations. For each control category, we synthesize two different results, one on the higher (e.g., "high left wrist position") and one on the lower (e.g., "low left wrist speed") end of each control signal value, defined by the 85th and 15th percentile of the training data. For our method, this effectively limits its search space to only 15% of the total training data. We also include the ground truth as the topline comparison. Each user was shown one video from each control level. The audio track is randomly sampled from one of six possible test sequences. The user study involved 42 respondents. Table 1 summarizes the result of the study. Our proposed k-NN+cGAN is consistently better at producing natural-looking and in-sync results compared to MoGlow. This result also suggests that our search-based approach can produce plausible synthesis even with a smaller search space produced by conditioning.

4.1.2 *Quantitative Evaluation.* Here we quantitatively analyze the performance of each method when subjected to a particular control signal. It should be noted that our method and MoGlow use the control signals in a different way to produce the desired outcome. Our k-NN-based algorithm achieves gesture control by using the control value as a threshold to limit the search space, while MoGlow directly use the control value as a regression target to modify a certain outcome of the gesture. Because of this, we treat the control signal differently for each method. To allow a higher gesture variation, we only enforce control on the first and last frame of the gesture candidate. While this allows the output to vary outside the specified threshold, this ensures the average value of the controlled variable will be close to the desired (threshold) value, while at the same time ensuring greater motion variability. In contrast, MoGlow uses the control input directly as a target value that needs to be satisfied in the output space. Therefore, their output gesture often produces less variation over the motion space, although it generally stays closer to the intended control signal. For example, if the left hand is conditioned to be at certain height, it is no longer able to produce body gestures with varying hand height. This, in many cases, makes the hand appear "stuck" at the given height. Correcting this would require significant manual labor. In contrast, our controlled results appear more realistic since combining real motion sequences from the database will likely induce natural modulation. Table 2 compares the predicted value with respect to the control signal of each method and Figure 2 shows the quantitative behavior of each method when conditioned by the given control signal. Please refer to the supplementary video for more qualitative comparisons.



(a) Wrist position using "high hand" control

(b) Wrist position using "fast speed" control

Figure 2: Control-based comparison of left hand height (a) and velocity (b) between k-NN (ours, blue), k-NN+cGAN (ours, orange), and MoGlow ([Alexanderson et al. 2020], green) over a test sequence. The larger variation produced by our methods lead to more natural motion variations, unlike MoGlow, which could lead to a temporally static gesture w.r.t. the control signal.

4.2 Synthesis with Complex and Low-level Control

Unlike MoGlow [Alexanderson et al. 2020], our gesture control can be achieved without model re-training. Hence, various control signals can be given during test time at any particular frame window, enabling the user to perform far more complex motion control. This is particularly useful when generating gestures that reflect the emotional state of the speaker. For example, if the speech of the speaker reflects an emotional change from sad to angry, we may want to synthesize gesture with slow and low hand position at the

Table 2: Quantitative comparison of control-based synthesis for left wrist height, speed, and symmetry. Our approach generates more natural looking gestures with larger motion variations. MoGlow, however, produces gestures with less variation which can be “stuck” at a given control signal, such as height, rendering unnatural-looking results.

Method	Threshold/ Target	Mean	Deviation/ Variation ↑
k-NN (wrist high)	22.2 cm	25.6 cm	3.4 cm
k-NN+cGAN (wrist high)	22.2 cm	23.4 cm	**4.2 cm**
MoGlow (wrist high)	22.2 cm	22.2 cm	1.1 cm
k-NN (wrist low)	9.7 cm	8.5 cm	2.1 cm
k-NN+cGAN (wrist low)	9.7 cm	9.1 cm	**2.9 cm**
MoGlow (wrist low)	9.7 cm	9.9 cm	0.6 cm
k-NN (wrist fast)	19.1 cm/s	22.8 cm/s	**12.0 cm/s**
k-NN+cGAN (wrist fast)	19.1 cm/s	17.5 cm/s	10.3 cm/s
MoGlow (wrist fast)	19.1 cm/s	11.6 cm/s	5.8 cm/s
k-NN (wrist slow)	3 cm/s	5.9 cm/s	3.8 cm/s
k-NN+cGAN (wrist slow)	3 cm/s	5.4 cm/s	**4.1 cm/s**
MoGlow (wrist slow)	3 cm/s	5.5 cm/s	3.0 cm/s
k-NN (asymm.)	10 cm	12.2 cm	3.0 cm
k-NN+cGAN (asymm.)	10 cm	10.4 cm	**3.9 cm**
MoGlow (asymm.)	10 cm	9.7 cm	1.6 cm
k-NN (symmetric)	0 cm	1.0 cm	1.2 cm
k-NN+cGAN (symmetric)	0 cm	2.1 cm	**2.0 cm**
MoGlow (symmetric)	0 cm	0.7 cm	0.5 cm

beginning, and progress towards fast and extended hand form at the end of the speech. Such a synthesis scenario can be achieved by our framework in one pass without requiring any re-training. On the other hand, pure learning-based controlled synthesis methods will fail in such tasks since producing gestures with different control signals (e.g. “speed" vs. “height" control) require different models with different training sets. In addition to the high-level control synthesis described above, our formulation can also be extended to follow time-specific signals, including signals with semantically meaningful information. As an example, we show that our framework can be used to produce a specific body gesture whenever a specific keyword is detected in the speech. The keywords can be inferred from speech by applying an off-the-shelf speech-to-text system to the input audio. When such a keyword is detected, instead of loading a gesture from the standard database, the gesture is selected from a separate database containing gestures which are semantically correlated with the keyword. Please refer to our supplementary videos for our text-based gesture control example results.

4.3 Synthesis Evaluation without Control Signals

We evaluate our approach when no control signal is applied by comparing the performance of both of our proposed components against the ground truth and four different baselines. We include two different versions of our proposed k-NN: one where the candidates are first selected based on pose similarity (pose pre-selected k-NN), and another one where the pre-selection is performed based on audio similarity (audio pre-selected k-NN). We also report our kNN-cGAN result which uses the pose pre-selected k-NN result as input.

Table 3: User study results assessing the performance between synthesis methods in the absence of control signals. Our proposed k-NN + cGAN outperforms other baselines both in terms of naturalness and synchronization.

Method	Naturalness ↑	Synchrony. ↑
Ground Truth	**6.26 ± 1.02**	**5.99 ± 1.02**
Mismatched audio-gesture	-	5.48 ± 1.34
Habibie et al. [2021]	5.79 ± 1.16	5.66 ± 1.14
MoGlow [Alexanderson et al. 2020]	4.84 ± 1.79	4.83 ± 1.65
k-NN pose-only similarity	4.57 ± 2.11	4.82 ± 1.93
Ours kNN Audio pre-selected	5.73 ± 1.13	5.50 ± 1.21
Ours kNN Pose pre-selected	5.55 ± 1.38	5.33 ± 1.34
Ours kNN+cGAN	**5.83 ± 1.26**	**5.82 ± 1.13**

Our first baseline is a simple k-NN that only predicts the next gesture based on the 3D pose similarity at every frame T without considering audio similarity. Next, we compare against randomly paired audio-gesture sequences. This baseline has been reported to perform strongly in previous studies [Kucherenko et al. 2021]. Another baseline we include is the recent GAN-based gesture regression approach by Habibie et al. [2021]. Finally, we compare our approach against Alexanderson et al. [2020].

We conducted a user study involving 41 different respondents using the same instructions discussed at the beginning of Sec. 4. During the study, each respondent was shown 16 different synthesis videos from two different random audio tracks sampled from a total of 15 possible tracks.

The result of the study is shown by Table 3. The gesture refinement results produced by our proposed k-NN+cGAN achieved the highest score in terms of synchronization and naturalness, including the prior state-of-the-art method of Habibie et al. [2021]. Moreover, unlike their approach, which directly predicts the gesture from the audio input, ours can follow various control signals and generate different gesture sequences given the same audio. Our method also performed better than the control-aware 3D gesture synthesis approach of Alexanderson et al. [2020] in terms of naturalness and synchronization. Overall, the results obtained by our proposed method show that it consistently outperforms the state-of-the-art at synthesizing both control-based as well as unconstrained gestures.

5 CONCLUSION AND LIMITATIONS

We presented a novel approach for controllable speech-driven body gesture synthesis. Our approach utilizes database search together with adversarial learning to produce natural and synchronized gestures. Compared to prior work, our technique offers more diverse manipulation and does not require re-training for every control signal. Results show that our approach outperforms the state-of-the-art both in terms of naturalness and audio-synchronicity even in the absence of control.

Our proposed approach has several limitations. We currently use hand-designed criteria for extracting and estimating feature similarity. Hence, future work can investigate using a learning-based approach for extracting and measuring such feature similarity, akin to Chung and Zisserman [2016]. Our proposed cGAN is currently not conditioned on the control signal, which may lead the result to deviate from the intended outcome, even though our experiments suggest that the deviation is tolerable. Another limitation of our search-based algorithm is the potentially expensive computation time compared to single-pass inference approaches of the purely learning-based counterparts. Since our method searches through the whole database to find the closest candidate for every frame window, the time complexity grows quadratically with the number of sequences in the database. A potential extension to remedy this issue is to train both stages of our method (k-NN and cGAN) in an end-to-end manner like the work of Holden et al. [2020]. However, unlike locomotion, gesture synthesis from speech is a more ambiguous problem, making it difficult to directly translate their approach into our domain.

ACKNOWLEDGMENTS

This work was supported by the ERC Consolidator Grant 4DRepLy (770784).

REFERENCES

Chaitanya Ahuja, Dong Won Lee, Yukiko I. Nakano, and Louis-Philippe Morency. 2020. Style Transfer for Co-speech Gesture Animation: A Multi-speaker Conditional-Mixture Approach. In *European Conference on Computer Vision (ECCV)*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (Eds.).

Simon Alexanderson, Gustav Eje Henter, Taras Kucherenko, and Jonas Beskow. 2020. Style-controllable speech-driven gesture synthesis using normalising flows. *Computer Graphics Forum* (2020).

Okan Arikan and D. A. Forsyth. 2002. Interactive Motion Generation from Examples. *ACM Trans. Graph.* (2002).

Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein Generative Adversarial Networks. In *Proceedings of the 34th International Conference on Machine Learning*.

Uttaran Bhattacharya, Elizabeth Childs, Nicholas Rewkowski, and Dinesh Manocha. 2021. Speech2AffectiveGestures: Synthesizing Co-Speech Gestures with Generative Adversarial Affective Expression Learning. In *Proceedings of the 29th ACM International Conference on Multimedia (MM '21)*. Association for Computing Machinery, New York, NY, USA.

Michael Büttner and Simon Clavet. 2015. Motion Matching. https://www.youtube.com/watch?v=z_wpgHFSWss&.

Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2017. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. In *CVPR*.

Justine Cassell. 2000. Embodied Conversational Interface Agents. *Commun. ACM* (2000).

Justine Cassell, Catherine Pelachaud, Norman Badler, Mark Steedman, Brett Achorn, Tripp Becket, Brett Douville, Scott Prevost, and Matthew Stone. 1994. Animated Conversation: Rule-Based Generation of Facial Expression, Gesture & Spoken Intonation for Multiple Conversational Agents. In *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*.

Justine Cassell, H. Vilhjálmsson, and T. Bickmore. 2001. BEAT: the Behavior Expression Animation Toolkit. In *Proceedings of SIGGRAPH 2001*. 477–486.

Chung-Cheng Chiu and Stacy Marsella. 2011. How to Train Your Avatar: A Data Driven Approach to Gesture Generation. In *Proceedings of the 10th International Conference on Intelligent Virtual Agents (IVA'11)*. 127–140.

Chung-Cheng Chiu and Stacy Marsella. 2014. Gesture Generation with Low-dimensional Embeddings. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems (AAMAS '14)*.

J. S. Chung and A. Zisserman. 2016. Out of time: automated lip sync in the wild. In *ACCV*.

Ylva Ferstl, Michael Neff, and Rachel McDonnell. 2019. Multi-Objective Adversarial Gesture Generation. In *Motion, Interaction and Games*.

Ylva Ferstl, Michael Neff, and Rachel McDonnell. 2020. Understanding the Predictability of Gesture Parameters from Speech and Their Perceptual Importance *(Proceedings of the International Conference on Intelligent Virtual Agents)*.

Ylva Ferstl, Michael Neff, and Rachel McDonnell. 2021. ExpressGesture: Expressive gesture generation from speech through database matching. *Computer Animation and Virtual Worlds* (2021).

S. Ginosar, A. Bar, G. Kohavi, C. Chan, A. Owens, and J. Malik. 2019. Learning Individual Styles of Conversational Gesture. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. 2017. Improved Training of Wasserstein GANs. In *Advances in Neural Information Processing Systems*.

Ikhsanul Habibie, Weipeng Xu, Dushyant Mehta, Lingjie Liu, Hans-Peter Seidel, Gerard Pons-Moll, Mohamed Elgharib, and Christian Theobalt. 2021. Learning Speech-driven 3D Conversational Gestures from Video. In *Proceedings of the International Conference on Intelligent Virtual Agents*.

Dai Hasegawa, Naoshi Kaneko, Shinichi Shirakawa, Hiroshi Sakuta, and Kazuhiko Sumi. 2018. Evaluation of Speech-to-Gesture Generation Using Bi-Directional LSTM Network. In *Proceedings of the 18th International Conference on Intelligent Virtual Agents*.

Daniel Holden, Oussama Kanoun, Maksym Perepichka, and Tiberiu Popa. 2020. Learned Motion Matching. *ACM Trans. Graph.* (2020).

Sergey Ioffe and Christian Szegedy. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Proceedings of the 32nd International Conference on Machine Learning*.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. Image-to-Image Translation with Conditional Adversarial Networks. *CVPR* (2017).

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.).

Lucas Kovar, Michael Gleicher, and FrÃ©dÃ©ric Pighin. 2002. Motion Graphs. In *Proceedings of SIGGRAPH '02*. San Antonio, TX.

Taras Kucherenko, Patrik Jonell, Sanne van Waveren, Gustav Eje Henter, Simon Alexandersson, Iolanda Leite, and Hedvig Kjellström. 2020. Gesticulator: A Framework for Semantically-Aware Speech-Driven Gesture Generation. In *Proceedings of the 2020 International Conference on Multimodal Interaction*.

Taras Kucherenko, Patrik Jonell, Youngwoo Yoon, Pieter Wolfert, and Gustav Eje Henter. 2021. A Large, Crowdsourced Evaluation of Gesture Generation Systems on Common Data: The GENEA Challenge 2020 *(IUI '21)*.

G. Lee, Z. Deng, S. Ma, T. Shiratori, S. Srinivasa, and Y. Sheikh. 2019. Talking With Hands 16.2M: A Large-Scale Dataset of Synchronized Body-Finger Motion and Audio for Conversational Motion Analysis and Synthesis. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*.

Jehee Lee, Jinxiang Chai, Paul S. A. Reitsma, Jessica K. Hodgins, and Nancy S. Pollard. 2002. Interactive Control of Avatars Animated with Human Motion Data. *ACM Trans. Graph.* (2002).

Jina Lee and Stacy Marsella. 2006. Nonverbal behavior generator for embodied conversational agents. In *Intelligent virtual agents*. Springer, 243–255.

Yongjoon Lee, Kevin Wampler, Gilbert Bernstein, Jovan Popović, and Zoran Popović. 2014. Motion Fields for Interactive Character Locomotion. *ACM Trans. Graph.* (2014).

Sergey Levine, Philipp Krähenbühl, Sebastian Thrun, and Vladlen Koltun. 2010. Gesture controllers. In *SIGGRAPH '10*.

Sergey Levine, Christian Theobalt, and Vladlen Koltun. 2009. Real-Time Prosody-Driven Synthesis of Body Language. In *ACM SIGGRAPH Asia 2009 Papers*.

Jing Li, Di Kang, Wenjie Pei, Xuefei Zhe, Ying Zhang, Zhenyu He, and Linchao Bao. 2021. Audio2Gestures: Generating Diverse Gestures from Speech Audio with Conditional Variational Autoencoders. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 11293–11302.

S. Mariooryad and C. Busso. 2012. Generating Human-Like Behaviors Using Joint, Speech-Driven Models for Conversational Agents. *IEEE Transactions on Audio, Speech, and Language Processing* (2012).

David McNeill. 2000. *Language and Gesture*. Cambridge University Press.

Mehdi Mirza and Simon Osindero. 2014. Conditional Generative Adversarial Nets. *CoRR* abs/1411.1784 (2014).

Vinod Nair and Geoffrey E. Hinton. 2010. Rectified Linear Units Improve Restricted Boltzmann Machines. In *ICML*.

Najmeh Sadoughi and Carlos Busso. 2019. Speech-driven animation with meaningful behaviors. *Speech Communication* 110 (2019), 90 – 100.

Alla Safonova and Jessica K. Hodgins. 2007. Construction and Optimal Search of Interpolated Motion Graphs. *ACM Trans. Graph.* (2007).

Eli Shlizerman, Lucio Dery, Hayden Schoen, and Ira Kemelmacher-Shlizerman. 2018. Audio to body dynamics. *CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.

Harrison Jesse Smith and Michael Neff. 2017. Understanding the impact of animated gesture performance on personality perceptions. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 1–12.

Youngwoo Yoon, Bok Cha, Joo-Haeng Lee, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. 2020. Speech Gesture Generation from the Trimodal Context of Text, Audio, and Speaker Identity. *ACM Transactions on Graphics* (2020).

Youngwoo Yoon, Woo-Ri Ko, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. 2019. Robots Learn Social Skills: End-to-End Learning of Co-Speech Gesture

Generation for Humanoid Robots. In *Proc. of The International Conference in Robotics and Automation (ICRA)*.

Youngwoo Yoon, Keunwoo Park, Minsu Jang, Jaehong Kim, and Geehyuk Lee. 2021. SGToolkit: An Interactive Gesture Authoring Toolkit for Embodied Conversational Agents. In *The 34th Annual ACM Symposium on User Interface Software and Technology (UIST '21)*. Association for Computing Machinery.