Improving Conditional-GAN using Unrolled-GAN for the Generation of Co-speech Upper Body Gesture

Bowen Wu[1,2]* Chaoran Liu[3] Carlos T. Ishi[2,3] Hiroshi Ishiguro[1,3]

[1] Osaka University
[2] RIKEN
[3] ATR

Abstract: Co-speech gesture is a crucial non-verbal modality for humans to express ideas. Social agents also need such capability to be more human-like and comprehensive. This work aims to model the distribution of gesture conditioned on human speech features for the generation, instead of finding an injective mapping function from speech to gesture. We propose a novel conditional GAN-based generative model to not only realize the conversion from speech to gesture but also to approximate the distribution of gesture conditioned on speech through parameterization. Objective evaluation show that the proposed model outperforms the existing deterministic model in terms of distribution, indicating that generative models can approximate the real patterns of co-speech gestures more than the existing deterministic model. Our result suggests that it is critical to consider the nature of randomness when modeling co-speech gestures.

1 Introduction

Human-like robots and virtual agents have human appearance. Therefore, they are expected to use both verbal and non-verbal behaviors to express themselves like humans during the interaction with humans. One crucial non-verbal behavior that can be observed is the hand gestures[1][2]. These spontaneous hand movements accompany speech to complement or even to supplement the information relayed in a speaker's speech[3]. Modeling the relationship between gestures and speech will provide a useful tool for expressing ideas comprehensively for human-like agents and promoting humans' perception.

At the early stage, robot gestures were only designed for a few pre-defined scenarios[4]. For the automatic generation, the first trial was the so-called ruled-based method. A set of human gesture patterns is recorded as sequences of joint data, and their occurrence was statistically studied in the relationship with the lexicon. These results were then summarized as a bunch of rules to decide which gesture to select from the recorded database[5]. An advanced rule-based method was proposed to separately model different parts of the human body to generate different combinations as a whole[6]. The shape of the gesture was constrained on those appearing in the collected data in these studies.

Beyond writing rules, data-driven statistical models were also adopted. The relation between iconic gestures and lexicon was automatically learned from the corpus using a Bayesian Decision Network[7]. Dynamic Bayesian Network was also utilized to model several meaningful behaviors (e.g., nod) while considering the synchronization with speech[8]. The relationship between the prosodic feature of speech and rhythmic gesture was modeled using modified hierarchical factored conditional restricted Boltzmann machines(HFCRBMs)[9]. Taking both prosody and text as input, a probabilistic model that maps the concept extracted from text using WordNet to gesture clusters, integrated with the superimposed beat gesture, was proposed[10]. However, the methods proposed in these studies require an elaborate feature engineering on human data.

Since human data analysis is tedious and time-consuming, machine learning and deep learning approaches have been utilized to automatically map speech to gesture. Hidden Markov Model was used to generate pointing gesture from audio features[11]. The effectiveness of recurrent models such as gated recurrent unit(GRU) and long-short term

*連絡先： 大阪大学大学院基礎工学研究科
〒 560-0043 大阪府豊中市待兼山町１丁目３
E-mail:wu.bowen@irl.sys.es.osaka-u.ac.jp

memory(LSTM) on mapping Mel-Frequency Cepstrum Coefficients(MFCC) features of speech to gestures has been analyzed[12][13]. In [13], a bi-directional LSTM network learned the mapping from MFCC features to 3D joint coordinates of the skeleton from the dataset collected using MOCAP toolkit. The text was also used as input to generate meaningful gestures by sequence to sequence neural networks[14]. In [15], the text was encoded using bidirectional encoder representations from transformers(BERT) to be concatenated with audio features to generate gesture sequences. Due to the high dimension characteristic of human motion, Denoising autoencoder(DAE) was employed to reduce the dimension of motion to help the neural network to generalize[16]. [17] made use of labeled gesture phase information to constrain the dynamic of generated gestures. The individual style was concerned with separately training different neural networks with L1 distance and discriminative loss on a particular person's data[18]. A style transfer model aiming at generating gestures with personal style for others' voice was also proposed[19].

The generation methods mentioned above are based on a strong assumption: the mapping from speech to gesture is injective, i.e., only one gesture can be generated by these models for one speech segment. On the contrary to this assumption, there are alternatives for almost any gestures. There are numerous examples of this phenomenon, such as using left or right or both hands, hand at different height and radius, and so forth. Additionally, a human may perform new gestures that have never been performed before for a particular speech. We treat this randomness as an essential nature of co-speech gestures. As a result, we aim to design a generative model to realize the randomness of co-speech gestures.

Relatively few studies have walked into this field. [20] uses MoGlow to generate gestures while controlling the height, radius, or speed by inputting a controlling variable, realizing the gestures' variation. However, they rely on a manipulated signal, whereas we leave this to be entirely random by sampling a random variable from a prior distribution.

Inspired by the success of generative adversarial nets(GAN) on generation tasks, we proposed a GAN-based generative model to realize the conversion from speech to gesture while preserving the randomness. To optimize the model, we designed a discriminator to give dynamic feedback on the generator results. Mode collapse, a common failure in GAN training, is minimized by using the algorithm of unrolled generative adversarial nets(Unrolled-GAN). We experimented with our model on a Japanese speech/gesture dataset. The objective evaluation showed that the proposed model can better approximate real gesture distribution. User studies also confirmed the proposed model's effectiveness and showed no significant difference between the generated results of the proposed model and the ground truth(original human motions).

The contributions of this work are three-folded: (1) We proposed a novel deep-learning-based generative model for co-speech gesture generation. (2) We proposed a strategy for changing gesture patterns by manipulating the randomly sampled vector, and improved the performance. (3) We confirmed that the proposed model outperformed the existing deterministic model through experiments.

2 Method

2.1 Problem Formulation

The notations used in the rest of this article are denoted as follows: for a speech segment with length T, the features extracted from the audio signal is $\mathbf{s} = [s_t]_{t=1:T}$. The sequence of the absolute position of each joint in the 3-dimensional space is $\mathbf{j} = [j_t]_{t=1:T}$, where $j_t = [x_t^i, y_t^i, z_t^i]_{i=1:K}$, K is the total number of joints. The problem of generating gesture from speech then can be defined as to parameterize a model G by a parameter set θ such that $\mathbf{j}^{(m)} = G_\theta(\mathbf{s}^{(m)})$. Furthermore, we aim to model the conditional distribution $X_\mathbf{j}$ conditioned on the distribution $X_\mathbf{s}$. To achieve this, a random variable z sampled from a normal distribution $N(0,1)$ will be taken as input by the model. Thus, the problem is to find a paramter set θ such that $p(\mathbf{j}|\mathbf{s}) = G_\theta(z|\mathbf{s}), \mathbf{j} \sim X_G, \mathbf{s} \sim X_\mathbf{s}, z \sim N(0,1)$. The error between the parameterized distribution and real distribution is defined as $d(p(\mathbf{j}|\mathbf{s})_{\mathbf{j}\sim X_G}, p(\mathbf{j}|\mathbf{s})_{\mathbf{j}\sim X_j})$ to optimize G_θ. A model D paramterized by ϕ will be optmized to be the measurement of this error.

2.2 Feature Extraction

Motion features. The motion data in the corpus is composed by joint rotation and offset of each joint. We used the protocol provided in [16] to convert the joint's rotation values to absolute position values(APV) in three-dimensional(3D) space to meet our problem established in section 2.2. As the movements are concentrated on the upper body part, we only used the upper body's APV as the training label.

Speech features. The speech features used in this work are the prosodic features. Prosodic features include fundamental frequency(f0), intensity, and their first derivative and second derivative, as they reflect the rhythm of speech. Although MFCC features are frequently used in automatic speech recognition(ASR), they are not preferred here because the extracted features will be used as conditions in model D. Low dimensional features are expected to yield better results than a high dimensional one since high-dimension conditions will drastically reduce the number of samples included in that condition. An opensource audio signal processing package Parselmouth was used to extract intensity and fundamental frequency from the speech signal. First, using a window size of 10 milliseconds and hop length of 5 milliseconds, 200 frames of every second feature are extracted. Then, every ten frames' feature are averaged to be 20 frames per second(fps) to match the fps of motion data.

2.3 Methodology

Fully-connected layer(FC). FC is essential in the deep learning area. It consists of a weight matrix $W \in \mathbb{R}^{m \times n}$,and a bias vector $b \in \mathbb{R}^n$, where m is the input's dimension to the FC, n is the output's dimension of the FC. The computation inside FC is defined as equation (1).

$$A = x \cdot W + b \tag{1}$$

where $x \in \mathbb{R}^m$, $A \in \mathbb{R}^n$.

Bidrectional Long-Short Term Memory. Bi-LSTM is a general solution for sequence data modeling. It utilizes the past and future information to compute the output for the current timing. Considering that past and future information in the speech has influences on the current motion, we used bi-LSTM to capture the information flowing through time.

Generative Adversarial Nets(GAN). The essence of GAN is a min-max game between a generator and a discriminator. While the discriminator is being optimized to recognize whether its inputs are sampled from real data or generated fake data by the generator, the generator is trying to deceive the discriminator by learning to generate data that resembles real data. This adversarial system will reach the nash equilibrium in the end after the generator learned to generate real data. Intuitively, this is equivalent to that the generator approximates the real data distribution. [21] confirmed this hypothesis by proving that the generator is trying to minimize the Jensen–Shannon divergence between the generated distribution and the real data distribution when the discriminator is optimal.

Conditional-GAN(CGAN). CGAN can generate an entity in a specific category[22]. It adds the same conditional labels for both generator and discriminator. Mathematically, the distribution to which the GAN's generator is trying to approximate is replaced by the conditional distribution conditioned on a specific category. [23] used CGAN to model head motion with speech as conditional input.

Unrolled-GAN. Mode collapse is a common failure in GAN training, i.e., the generator outputs identical results for any noise vector from the prior. By unrolling the discriminator, unrolled-GAN allows the generator to "look into the future" to prevent the discriminator from overfitting on a specific training sample, reducing the mode collapse[24].

Proposed method. Our proposed model utilizes the architecture of CGAN, where the speech features are used as a condition. An overview is shown in Figure 1 and 2. During the generating phase, a randomly sampled vector(noise vector) z from the Gaussian prior is repeated to the time step length of speech features. Then, z and the speech feature s are concatenated and fed into a two-layer bi-LSTM. A sequence-wise fully-connected layer then takes the output of previous layers and outputs a sequence of vectors indicating each joint's absolute positions in the 3D space. The reason for repeating a fixed-length random vector instead of sampling a sequence length wise random vector is that we want to maintain the output motion's consistency along the entire sequence. To optimize the generator, we optimize a discriminator simultaneously to compute the error between the generated distribution and the real distribution conditioned on speech features. A vector of motion sequence and the corresponding speech features are concatenated and fed into a two-layer bi-LSTM layer. The output is squashed between 0 and 1 through a sigmoid function, indicating whether the input motion is real and corresponding with the speech features. Instead of outputting only one scalar for the whole sequence by the discriminator, we prefer to output one scalar for each time step. The reason for doing so is that though

LSTM is claimed to be capable of capturing long term dependencies, in practice, the effectiveness decreases when the sequence grows relatively long. The equation for optimizing generator and discriminator is defined as equation (2).

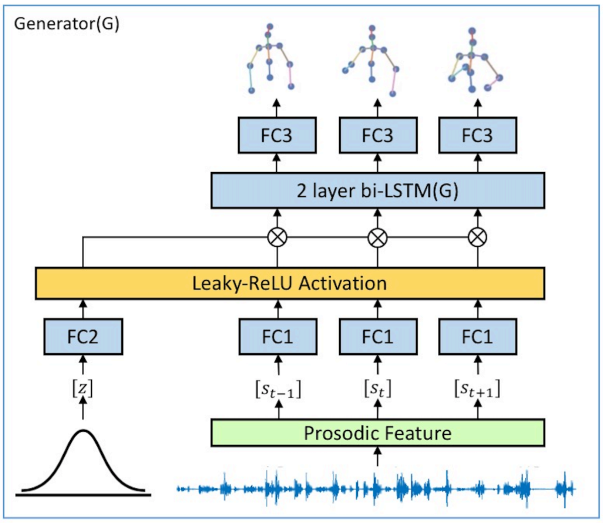

Fig. 1: The generator of proposed model. The output of FC2 is manipulated to the same time steps with **s**, and then be concatenated.

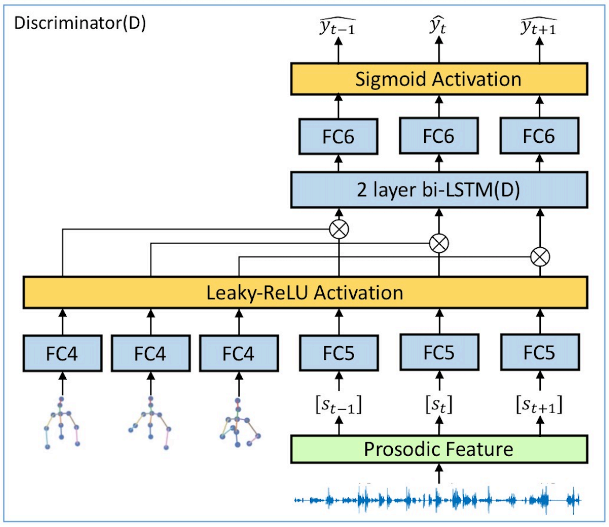

Fig. 2: The discriminator of proposed model. The outputs of FC4 is of the same length as **s**. The concatenation follows the order of the sequence.

$$\max_D \min_G \frac{1}{m}\sum_{i=1}^{m} log(D(j^{(i)}, s^{(i)})) - log(D(G(z, s^{(i)}), s^{(i)})) \tag{2}$$

where m is the number of samples.

Algorithm 1 Algorithm for training.

```
Require: $\alpha$, the learning rate. $k_{unroll}$, the unrolling steps. $m$, the batch size. $iteration$, the number of training iterations.
Require: $\phi_0$, initial discriminator parameters. $\theta_0$, initial generator parameters.
for 0 to $iteration$ do
    Sample $\{\mathbf{j}^{(i)}, \mathbf{s}^{(i)}\}_{i=1}^{m} \sim (X_{\mathbf{j}}, X_{\mathbf{s}})$ a batch from the real data
    Sample $\{z^{(i)}\}_{i=1}^{m} \sim N(0,1)$ a batch from the prior
    $g_\phi \leftarrow \nabla_\phi[\frac{1}{m}\sum_{i=1}^{m} log(D_\phi(j^{(i)}|s^{(i)})) - log(D_\phi(G_\theta(z^{(i)}, s^{(i)})|s^{(i)}))]$
    $\phi \leftarrow \phi + \alpha \cdot g_\phi$
    $backup_\phi \leftarrow \phi$
    for 0 to $k_{unroll}$ do
        Sample $\{\mathbf{j}^{(i)}, \mathbf{s}^{(i)}\}_{i=1}^{m} \sim (X_{\mathbf{j}}, X_{\mathbf{s}})$ a batch from the real data
        Sample $\{z^{(i)}\}_{i=1}^{m} \sim N(0,1)$ a batch from the prior
        $g_\phi \leftarrow \nabla_\phi[\frac{1}{m}\sum_{i=1}^{m} log(D_\phi(j^{(i)}|s^{(i)})) - log(D_\phi(G_\theta(z^{(i)}, s^{(i)})|s^{(i)}))]$
        $\phi \leftarrow \phi + \alpha \cdot g_\phi$
    end for
    Sample $\{\mathbf{s}^{(i)}\}_{i=1}^{m} \sim X_{\mathbf{s}}$ a batch from the real data
    Sample $\{z^{(i)}\}_{i=1}^{m} \sim N(0,1)$ a batch from the prior
    $g_\theta \leftarrow \nabla_\theta[\frac{1}{m}\sum_{i=1}^{m} log(D_\phi(G_\theta(z^{(i)}, s^{(i)})|s^{(i)}))]$
    $\theta \leftarrow \theta - \alpha \cdot g_\theta$
    $\phi \leftarrow backup_\phi$
end for
```

On the other hand, a common failure during GAN training is mode collapse, i.e., the generator outputs identical results for any noise vector from the prior. In practice, we found that the algorithm proposed in unrolled-GAN successfully reduced the mode collapse that appeared in our experiment setting. However, since we used the LSTM layer, the original unrolled-GAN algorithm will tremendously increase the training time. As a result, we simplified the algorithm and observed that a similar result is achieved in our experiment with shorter training time. Note that we are not claiming that the original algorithm is replaceable by this simplified version. The proposed algorithm is shown in Algorithm 1.

In our experiment, we found that each noise vector corresponds with a particular pattern of motion, i.e., motions with the same pattern are generated when using the same noise vector throughout the sequence, a result that is not desirable. To increase variations of the generated motions, we proposed a strategy of generating variating noise vectors for a certain length of speech sequence. The algorithm is shown in Algorithm 2.

Algorithm 2 Algorithm for generating noise vectors.

```
Require: $T$, time steps of speech features. $F$, time steps of repeating the same noise vector.
$K \leftarrow \text{ceil}(T/F)$
$zs \leftarrow [z, ..., z]_F \sim N(0, 1)$
for 0 to $K$ do
    Sample $P \sim Uniform(0, 1)$
    if $P > 0.5$ then
        append $zs_{:-F}$ to $zs$
    else
        $zs_1 \leftarrow [z_1, ..., z_1]_F \sim N(0, 1)$
        append $zs_1$ to $zs$
    end if
end for
```

3 Experiment and results

3.1 Corpus

We evaluated our model on the dataset proposed in [25], in which pairs of recorded audio and motion are provided. The content is an undergraduate student answering questions in Japanese like in an interview while standing and gesturing. The motion data was recorded using a motion capture studio. The motion data files contain information of offset and rotation of each joint, from which each joint's absolute position can be derived. The audio is saved as WAV files(sampling rate 22050 Hz, 16bits). There are 1049 sentences in this dataset, 298 minutes, 68.41% are metaphoric gestures, 23.73% are beat gestures, and others are iconic and deictic gestures.

3.2 Baseline

To compare the proposed model with the deterministic generation method, the model proposed in [16] is used as a baseline. We use the protocol provided by the authors and reproduced the reported result. We cut the upper body motion generated using the baseline model in order for comparison. Since the dataset is already split for the baseline model into train, development, and test set, we use the split test set for evaluation. There are 45 samples in the test set.

3.3 Training setting

Numerous works for gesture generation cut gesture sequence into several slices to approximate the effect of data augmentation. Instead, we used the entire sequence of speech and motion as samples. We saved the trained model with every ten iterations and generated some samples using speech utterances in the test set. By viewing the quality of these generated results, we finally chose the generator of the 1000 iteration.

3.4 Quantitive Evaluation

Numerical evaluation metric. It is common for a deterministic model to use L1 distance or average position error(APE) to evaluate the generated results. Since our motivation is to model the distribution of gestures, it is not appropriate to evaluate the precision of generated key-points compared with the ground truth. Instead, kernel density estimation(KDE) is a useful tool for approximating the distribution of data, as also used in [21] for image generation and in [23] for head motion generation. The output of KDE is the log-likelihood of input samples based on the fitted density function using samples. In this work, we use generated gesture sequences in the test set to fit the density function and use the ground truth as input to KDE. Therefore, the larger the output value towards 0, the similar the generator fits the real data distribution.

By using the algorithm 2, we generated one motion sequence for every speech in the test set. The generated motions are used to fit a distribution. The optimal bandwidth in the KDE model is obtained using a grid search with 3-fold cross-validation. Then, the log-likelihood of real motions in the test set is calculated using the fitted distribution. We also studied how F in the algorithm 2 affects the results. The results are shown in Table 1. The values are the average of 5 times calculation.

Table. 1: Quantitive comparison between models. Ground Truth is the log-likelihood of real motions in the test set in the KDE distribution fitted using the ground truth itself, indicating the best results that can be approached. * used repeated noise vector to generate motions. ** jointly used the proposed model and the proposed algorithm 2.

Model	Log-likehood	SE
Ground Truth	*-29.98*	*1.03*
Baseline[16]	-508.82	87.61
CGAN*	-245.67	44.72
Unrolled-CGAN*	-118.91	17.03
Proposed(F=20)**	-177.86	29.36
Proposed(F=30)**	-161.30	26.78
Proposed(F=40)**	**-107.58**	**15.21**
Proposed(F=50)**	**-107.98**	**15.77**
Proposed(F=60)**	-126.20	19.01

4 Discussion

4.1 Why Generative Models Perform Better?

Modeling the distribution. L1-distance or L2-distance are usually used as the loss function for training deterministic models. A potential risk of doing so is that if there are two similar speech utterances paired with different gestures in the training set, an average value of these gestures will be the optimal solution for these utterances, and this average value is likely not being an available gesture at all. This risk may either cause the generated motions to be not human-like or small range gestures. On the other hand, generative models do not have such a problem since generative models aim to approximate the likelihood of real data. The generated gestures are more reliable and have the potential to cover a broader range of gestures. This difference can be seen from the comparison between baseline and other generative models in Table 1.

4.2 Detailed performance analysis

Motion dynamics(i.e., velocity) are imperative to human perception. Since we are aiming at modeling the distribution of human gesture, one reason that the proposed model outperforms the baseline model is assumed to be that the velocity distribution of the motion generated by the proposed model is more similar with the ground truth than the baseline model. By plotting the histogram of average velocity of all joints, shoulder, and wrist, we confirmed this assumption by observing that the histograms of proposed model are more similar with the ground truth compared with the baseline, as shown in Fig. 3, 4, and 5.

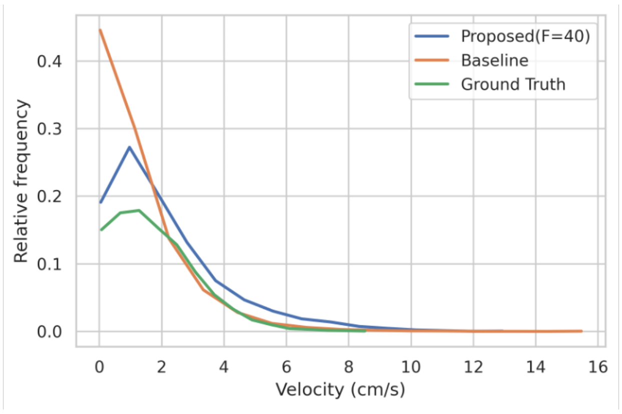

Fig. 3: Histogram of average velocity of all joints.

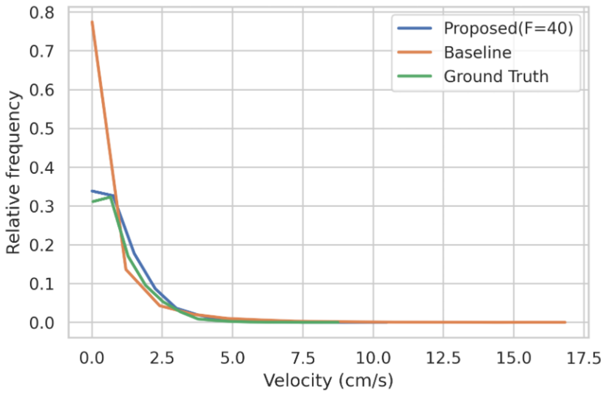

Fig. 4: Histogram of shoulder velocity of all joints.

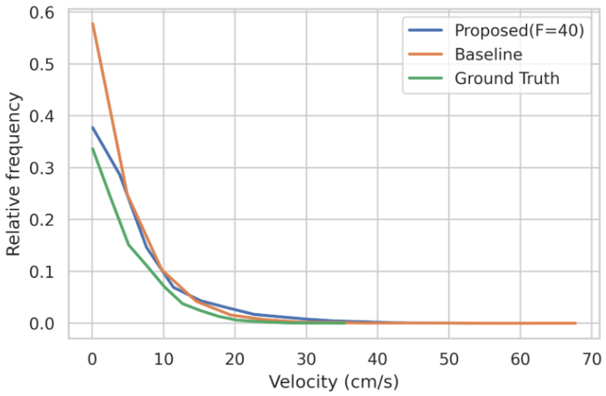

Fig. 5: Histogram of wrist velocity of all joints.

5 Conclusions

It is crucial for human-like agents to gesture well to be comprehensive and expressive. We present a model for producing co-speech gestures by modeling the conditional distribution of gesture conditioned on speech features. Improved by unrolled-GAN and our proposed algorithm, the proposed model outperforms the existing deterministic model in objective evaluation. Our work provides a powerful tool for Human-like agents to express thoughts, enhancing human perception. Moreover, probabilistic modeling's success reveals that future research in this field should focus more on gesture distribution. Human-like agent is expected to realize complicated interaction with human. However, without the ability to gesture well, they are inexpressive to be understood or empathized with by humans. Though our gesture generation model performs better in terms of gesture distribution, the lack of semantics(i.e., meaningful gesture) is still a considerable obstacle to perfectly model human gesture, which requires further research.

Acknowledgement

This work was supported by Grant-in-Aid for Scientific Research on Innovative Areas JP20H05576.

References

[1] Adam Kendon. *Gesture: Visible action as utterance.* Cambridge University Press, 2004.

[2] David McNeill. *Gesture and thought.* University of Chicago press, 2008.

[3] Jana M Iverson and Susan Goldin-Meadow. Why people gesture when they speak. *Nature*, Vol. 396, No. 6708, pp. 228–228, 1998.

[4] Justine Cassell. A framework for gesture generation and interpretation. *Computer vision in human-machine interaction*, pp. 191–215, 1998.

[5] Björn Hartmann, Maurizio Mancini, and Catherine Pelachaud. Implementing expressive gesture synthesis for embodied conversational agents. In *International Gesture Workshop*, pp. 188–199. Springer, 2005.

[6] David Baumert, Shunsuke Kudoh, Masaru Takizawa, et al. Design of conversational humanoid robot based on hardware independent gesture generation. *arXiv preprint arXiv:1905.08702*, 2019.

[7] Kirsten Bergmann and Stefan Kopp. Gnetic–using bayesian decision networks for iconic gesture generation. In *International Workshop on Intelligent Virtual Agents*, pp. 76–89. Springer, 2009.

[8] Najmeh Sadoughi and Carlos Busso. Speech-driven animation with meaningful behaviors. *Speech Communication*, Vol. 110, pp. 90–100, 2019.

[9] Chung-Cheng Chiu and Stacy Marsella. How to train your avatar: A data driven approach to gesture generation. In *International Workshop on Intelligent Virtual Agents*, pp. 127–140. Springer, 2011.

[10] Carlos T Ishi, Daichi Machiyashiki, Ryusuke Mikata, and Hiroshi Ishiguro. A speech-driven hand gesture generation method and evaluation in android robots. *IEEE Robotics and Automation Letters*, Vol. 3, No. 4, pp. 3757–3764, 2018.

[11] Mehmet Emre Sargin, Oya Aran, Alexey Karpov, Ferda Ofli, Yelena Yasinnik, Stephen Wilson, Engin Erzin, Yücel Yemez, and A Murat Tekalp. Combined gesture-speech analysis and speech driven gesture synthesis. In *2006 IEEE International Conference on Multimedia and Expo*, pp. 893–896. IEEE, 2006.

[12] Ylva Ferstl and Rachel McDonnell. Investigating the use of recurrent motion modelling for speech gesture generation. In *Proceedings of the 18th International Conference on Intelligent Virtual Agents*, pp. 93–98, 2018.

[13] Dai Hasegawa, Naoshi Kaneko, Shinichi Shirakawa, Hiroshi Sakuta, and Kazuhiko Sumi. Evaluation of speech-to-gesture generation using bi-directional lstm network. In *Proceedings of the 18th International Conference on Intelligent Virtual Agents*, pp. 79–86, 2018.

[14] Youngwoo Yoon, Woo-Ri Ko, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. Robots learn social skills: End-to-end learning

of co-speech gesture generation for humanoid robots. In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 4303–4309. IEEE, 2019.

[15] Taras Kucherenko, Patrik Jonell, Sanne van Waveren, Gustav Eje Henter, Simon Alexanderson, Iolanda Leite, and Hedvig Kjellström. Gesticulator: A framework for semantically-aware speech-driven gesture generation. *arXiv preprint arXiv:2001.09326*, 2020.

[16] Taras Kucherenko, Dai Hasegawa, Gustav Eje Henter, Naoshi Kaneko, and Hedvig Kjellström. Analyzing input and output representations for speech-driven gesture generation. In *Proceedings of the 19th ACM International Conference on Intelligent Virtual Agents*, pp. 97–104, 2019.

[17] Ylva Ferstl, Michael Neff, and Rachel McDonnell. Multi-objective adversarial gesture generation. In *Motion, Interaction and Games*, pp. 1–10. 2019.

[18] Shiry Ginosar, Amir Bar, Gefen Kohavi, Caroline Chan, Andrew Owens, and Jitendra Malik. Learning individual styles of conversational gesture. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3497–3506, 2019.

[19] Chaitanya Ahuja, Dong Won Lee, Yukiko I Nakano, and Louis-Philippe Morency. Style transfer for co-speech gesture animation: A multi-speaker conditional-mixture approach. *arXiv preprint arXiv:2007.12553*, 2020.

[20] Simon Alexanderson, Gustav Eje Henter, Taras Kucherenko, and Jonas Beskow. Style-controllable speech-driven gesture synthesis using normalising flows. In *Computer Graphics Forum*, Vol. 39, pp. 487–496. Wiley Online Library, 2020.

[21] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

[22] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[23] Najmeh Sadoughi and Carlos Busso. Novel realizations of speech-driven head movements with generative adversarial networks. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6169–6173. IEEE, 2018.

[24] Luke Metz, Ben Poole, David Pfau, and Jascha Sohl-Dickstein. Unrolled generative adversarial networks. *arXiv preprint arXiv:1611.02163*, 2016.

[25] Kenta Takeuchi, Souichirou Kubota, Keisuke Suzuki, Dai Hasegawa, and Hiroshi Sakuta. Creating a gesture-speech dataset for speech-based automatic gesture generation. In *International Conference on Human-Computer Interaction*, pp. 198–202. Springer, 2017.