Probabilistic Human-like Gesture Synthesis from Speech using GRU-based WGAN

ABSTRACT

Gestures are crucial for increasing the human-likeness of agents and robots to achieve smoother interactions with humans. The realization of an effective system to model human gestures, which are matched with the speech utterances, is necessary to be embedded in these agents. In this work, we propose a GRU-based autoregressive generation model for gesture generation, which is trained with a CNN-based discriminator in an adversarial manner using a WGAN-based learning algorithm. The model is trained to output the rotation angles of the joints in the upper body, and implemented to animate a CG avatar. The motions synthesized by the proposed system are evaluated via an objective measure and a subjective experiment, showing that the proposed model outperforms a baseline model which is trained by a state-of-the-art GAN-based algorithm, using the same dataset. This result reveals that it is essential to develop a stable and robust learning algorithm for training gesture generation models.

KEYWORDS

gesture generation, social robots, generative model, neural network, deep learning

ACM Reference Format:

. 2018. Probabilistic Human-like Gesture Synthesis from Speech using GRU-based WGAN. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Human-like agents can play various roles in this society, such as guidance, receptionist, TV hosts, presenter, and so on. These agents have human appearances, so that they are expected to behave like humans. During human interaction, gesture is one crucial non-verbal behavior that may convey different semantic as well as affective information. Thus, a system for generating natural gestures is necessary for an embodied or virtual human-like agent to fill the gap with humans.

There are various ways for generating gestures. For specific expressions, it can be hand-crafted by creating the movements on an agent by directly editing the control commands [2]. Mapping human motion data with word concepts through gesture functions and dialogue act information is also possible [11][12][17]. However, these methods need expert's knowledge and require extensive analysis of data. On the other hand, automatically creating generation models using learning algorithms is straightforward, if a large-scale data is available. In this work, we aim at developing learning algorithms to train a model that can generate natural gestures, based on large-scale data.

Numerous deep learning-based models have been proposed to model human gestures. Even though the probabilistic models (i.e., models that can produce multiple outputs for one input) outperform deterministic models (i.e., models that produce single output for one input), the naturalness of the generated motion is still not achieving the human level [22]. Thus, in this work, we aim on building a better model for the generation of gestures. We proposed a novel deep-learning-based model for gesture generation and successfully trained it using the loss function we designed. We designed a complete synthesis protocol for gesture generation, which outputs the rotation vector of the upper body joints. We confirmed that the synthesized motions of the proposed model outperform the baseline model via objective and subjective measures.

The rest of this article is organized as follows. In section II, we present the studies related to the present work. Section III provides a complete explanation of the proposed system. In section IV, the experiment for evaluating the proposed system is detailed. In section V, the obtained results are discussed.

2 RELATED WORK

2.1 Speech-Driven Gesture Synthesis via Learning

Recently, deep learning methods have been widely used on the gesture generation task because several open-access human gesture databases have been developed in these years [21][5]. Deterministic models assume that there is only one gesture corresponding to one segment of audio. For example, [9] used LSTM (long short term memory) to realize the mapping from MFCC (Mel-Frequency Cepstral Coefficient) features extracted from the input audio to gestures. [14][15] analyzed how different features of audio input affect the result, and proposed to use low-dimensional manifold as training target instead of high-dimensional raw data. A style transfer model was developed to generate gestures with personal trait using arbitrary audio input (i.e., different person's voice) [1]. Text of the speech was also included in the inputs to the model to generate gestures [16][23]. However, these models are trained using mean squared error (MSE) as loss function so that they can only produce one motion sequence for a given audio segment input.

On the other hand, since there can be multiple solutions of gesture sequence for one audio segment, probabilistic modeling of human gestures has been proposed. These models can produce different available motion sequences for one audio segment input. One approach is by using Glow-based model, which maps the source distribution to the target distribution [13]. For instance, MoGlow was used to model the distribution of human gestures [3]. In this model, one audio segment can be used to generate multiple motion sequences because the loss function for training MoGlow is

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Woodstock '18, June 03–05, 2018, Woodstock, NY

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

the likelihood of the rotation values of each joint in the real data distribution. Another direction is to utilize generative adversarial network (GAN) [24][22]. The probabilistic generation is realized by inputting a randomly sampled noise vector to the generator. Different noise vectors can lead the model to produce different motions. A common failure, called mode collapse, in GAN training can be reduced by using the learning algorithm of unrolled-GAN[22]. However, Glow-based models have a huge number of parameters to learn, which is known as being hard to train. The generated motions of the previously mentioned GAN model appears to move too much, which negatively affects the naturalness. The reason is likely to be that the model only partially covers the real motion distribution, failing to learn the part of the real motion distribution with less or no movements. In other words, due to the learning algorithm of the model, mode collapse is still happening and severely damages the performance of the model.

2.2 WGAN-GP

Although vanilla-GAN has a common issue that the model does not converge so that the training is unstable and it is hard to diagnose in order to improve the performance, the learning algorithm of Wasserstein-GAN (WGAN) is supposed to be effective for training GAN model, because it is mathematically convergent, and thus provides a reliable indication of the progress of the training process [4]. WGAN has shown its success in multiple research areas including speech synthesis[25], music generation[18], and natural language modeling[19].

Gradient penalty is an advanced technique to stabilize the training process of GAN. It penalizes the norm of the gradients on the parameters of the discriminator model to ensure Lipschitz continuity [8].

In this work, we adopted the learning algorithm of WGAN-GP (WGAN with gradient penalty), to train a motion generator for the modeling of human gesture. The consideration is that with an advanced training algorithm to reduce the effect of mode collapse, the model can approximate the real motion distribution much better, thus generating more natural and human-like gestures.

3 PROPOSED GESTURE SYNTHESIS SYSTEM

In the proposed system, we formulate the gesture generation problem as modeling conditional distribution, given the observed speech. Specifically, given parameters extracted from a segment of speech, we try to sample corresponding gesture of same period from the conditional distribution modeled by a deep neural network. In addition, we employ Huber cost during the training process to explicitly reduce the discontinuity within the sampled gestures.

3.1 Data Pre-processing

Prosodic features are extracted from the speech signal to be used as conditional parameters for the GAN model. Specifically, voice pitch and power values were estimated each 10ms [10]. For the voice pitch features, the F0 (fundamental frequency) values were estimated by a conventional autocorrelation-based method. All the estimated F0 values were then converted to a musical (log) scale in semitone intervals before any subsequent processing. The power values were computed in dB units.

Although the motion data used in this work includes joints both in the upper body and lower body, which is recorded by Motion Capture Toolkit [21], only the upper body joints were used as target of modeling, since most movements in the dataset are concentrated on the upper body. As a result, the data of 12 joints are selected to form the training data.

3.2 Gesture Generator Architecture

An overview of the proposed system is shown in Fig. 1. First, the prosodic feature vector is extracted from the audio segment. Then, it is concatenated with a randomly sampled noise vector and fed into the generator. Through computation, the generator outputs frames of rotation vectors. The discriminator will be discussed in the following sub-section. Additionally, to avoid jerky motion, a low-pass filter is used to post-process the generated rotation vectors.

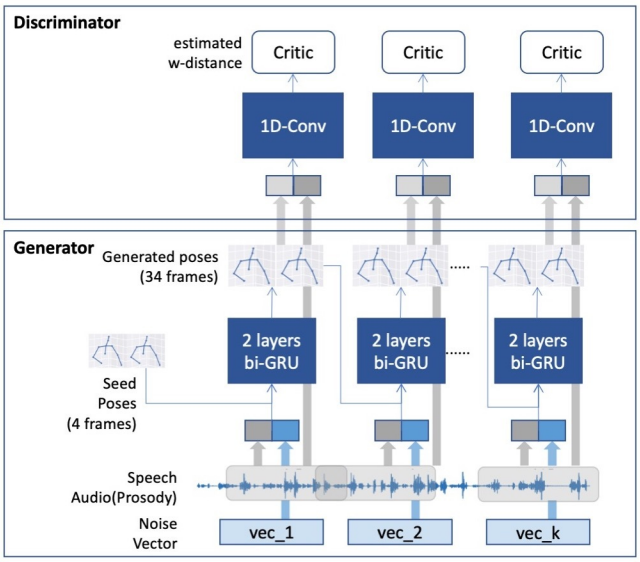

Figure 1: An overview of the proposed system.

The generator consists of a prosodic feature extractor and a 2-layer bi-GRU (bi-directional gated recurrent unit) network. The bi-GRU network takes its input as the concatenation of the extracted audio features, the randomly sampled noise vector, and seed poses. The audio is segmented into 1.5 seconds with an overlap of 0.2 seconds. For each audio segment, 34 frames of motion are produced. Overlap of motion is interpolated using 4 frames at the last of the previous motion chunk, and 4 frames at the beginning of the next motion chunk. The discriminator composes a 1-dimensional convolution layer, which takes input as the concatenation of the motion chunk and audio segment and outputs a scalar to indicate the distance between the real data distribution and the generated distribution.

3.3 Training

First of all, in order to realize a probabilistic generation, a discriminator is used to train the generator by an adversarial training.

Secondly, the motion data in the dataset are re-shaped to short chunks (1.5 seconds for one chunk). This is by considering that a long sequence of motion can be divided into several short motion chunks. If the motions in these chunks are natural, and the transition between these chunks are also natural, the whole sequence of motion will be natural enough. This way, we can focus on training the generator to generate each short motion chunk and to realize the transition between these chunks, instead of producing a whole sequence of motions, which is more difficult for learning. This brings us to use a convolutional layer based discriminator, as the length of each training sample is relatively short. Additionally, the input to the discriminator includes the extracted audio features to force the generated motion to be synchronized with the input audio. As a result, the loss for training the generator consists of three parts: the critic loss provided by the discriminator, the continuity loss for the transition between motion chunks, and the gradient penalty loss for training stability.

$$\mathcal{L}_{total} = \mathcal{L}_{critic} + \lambda_{gp} * \mathcal{L}_{gp} + \lambda_c * \mathcal{L}_{continuity} \tag{1}$$

3.3.1 Critic Loss. Critic loss is the traditional loss function of WGAN [4]. The critic loss is computed by an optimal discriminator which outputs the distance between the conditional distribution of the generated rotation vector and that of the real rotation vector, conditioned on the input audio features. The optimal discriminator is approximately obtained by previously training the discriminator to maximize the distance between the two conditional distributions. Then, the calculated distance is used as the loss for training the generator through back-propagation. The critic loss is defined as

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

where y represents rotation vector, s represents audio features, z represents random sampled vector, D is the discriminator, G is the generator.

3.3.2 Gradient Penalty (GP). Gradient penalty is a regularization term for achieving stable training [8]. It punishes the norm of the gradients on the discriminator to be equal to 1, which requires to calculate the derivatives of the output of the discriminator with respect to its inputs. We applied the learning algorithm of WGAN-GP, in order to achieve a more stable training for our gesture generation model. Since our conditional discriminator has double inputs, we compared the result of only punishing the norm of the gradients based on one of them (only the part connected with the generator), and all of them. It turned out that there was no significant difference. Thus, under the current training setting, punishing only one side seems enough and time-saving. The gradient penalty loss is defined as

$$\mathcal{L}_{gp} = \frac{1}{m} \sum_{i=1}^{m} \|\nabla_D^{(i)}\|_{L2} \tag{3}$$

3.3.3 Continuity Loss. Since the generator can only generate motion chunks whose length is 1.5 seconds due to the training settings, continuity loss is used to force the frames at the beginning of the next motion chunk to be similar with those at the end of the previous motion chunk. This way, these motion chunks can be concatenated directly to form a whole motion sequence. The continuity loss is defined as

$$\mathcal{L}_{continuity} = \frac{1}{m} \sum_{i=1}^{m} \text{Huber}(y_{:k}^{(i)}, y_{:-k}^{(i-1)}) \tag{4}$$

where k is a hyper-parameter to define how many frames to force to be continuous with the previous motion chunk, and the Huber loss is defined in [6].

3.3.4 Training procedure. We train the proposed model on a Japanese dataset, in which the motion data is recorded with Motion Capture Toolkit. The data contains 1094 pairs of motion and audio. The total length of the data is 6 hours. To train the model, the learning rate was set to 10^{-4} for both the generator and discriminator. Batch size is set to 128. Lambda for continuity loss is set to 1. Lambda for GP is set to 10. The distribution for sampling noise vector is a Gaussian with 0 mean and one variance. The model was trained on RTX Titan GPU, for approximately 6 hours.

4 EVALUATION

It is not appropriate to evaluate motions using objective measures such as average position error for value-level comparison, because a motion can look natural even if the score is low based on these measures [23]. Kernel density estimation (KDE) is reported to be an appropriate measure for evaluating distributions as it outputs the log-likelihood of one distribution in another distribution, as used in [20][7]. Thus, we use KDE to objectively evaluate the generated results of the proposed model. Another common method for evaluating motions is based on user-study, which utilizes human perception. We conducted a user-study to assess the performance of the proposed model. The rotation values of joints is not straightforward for humans, but they can be visualized through motions in an avatar. In this work, we implemented motions on a virtual avatar using the Unity software (Fig. 2).

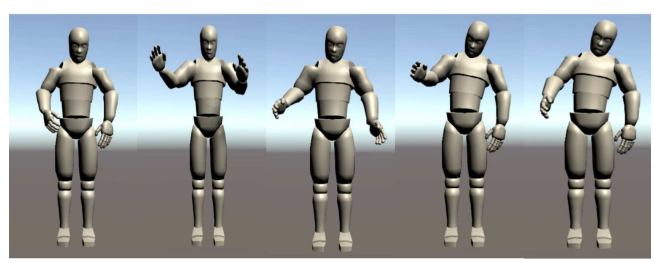

Figure 2: Snapshots of upper-body motions synthesized in the avatar.

We compare the generated motion of the proposed model with different control groups using video clips. The details are shown below:

- **Ground Truth (GT)**. Real data recorded from human using Motion Capture Toolkit. Before applied to the avatar, a low-pass filter is used to pre-process (smooth) the data to avoid jerk motion.
- **Baseline**. The baseline model we chose is a state-of-the-art model for generating coordinate values of joints, trained on the same dataset used for training the proposed model. This model is a GAN-based probabilistic gesture generation model proposed in [22]. In order to make comparison, we trained a neural network to convert the coordinate values to joint rotation values for the corresponding joints. The training data is the data used to train the proposed model.
- **Proposed Model**. As the goal is to realize probabilistic generation, the model is able to produce multiple motion sequences for one audio segment. To assess the result of different generated samples, we prepare two videos for the proposed model within each utterance set. Note that they are synthesized using different noise vectors for the same audio input.

4.1 Objective Evaluation

The motions generated using the proposed model and baseline model were used to fit a KDE model. The optimal bandwidth was obtained using a grid search with 3-fold cross-validation. Then, the log-likelihood of the real motion data in the test set was calculated using the fitted KDE model. Thus, as the output value tends to be larger, the motions better fit the real data distribution. A comparison between different models is shown in Table 1. The result of GT (Ground Truth) is calculated by computing the likelihood of the ground truth data using the KDE model fitted by the ground truth data itself. This can be seen as the best result that can be achieved. As the log-likelihood of a model tends to be similar with that of GT, the model will better fit the GT data distribution.

Table 1: KDE evaluation results for different models.

Model	Log-likelihood	Standard Error
GT	-122.53	0.23
Baseline	-152.37	25.26
Proposed	-125.60	0.10

It can be observed from the results that the proposed model achieves log-likelihood valued closer to the GT data.

4.2 Subjective Evaluation

There are three aspects that were measured for evaluating the proposed model, which are naturalness of the motion, the time consistency between the motion and the input speech, and their semantic connection. For that purpose, we used the question settings used in [14][22].

We compare the generated motion of the proposed model with different control groups using video clips. The details are follows.

The expected result is that the gestures generated by the proposed model are assigned better scores than the baseline model, and are approaching the level of the ground truth for all aspects in the questionnaires. Totally, we prepared 5 sets for 5 different utterances, in which there are 4 videos within each set (GT, Baseline, and two for the proposed model). The order of the videos is randomized. Participants are required to assign scores (from 1 to 7, 1 represents negative, 7 represents positive) for each scale of the gesture performed by the avatar defined in table 2 after watching each video.

Table 2: Likert scale questions used in the user study.

Scale	Statements (Translated from Japanese)
Naturalness	Gesture was natural
	Gesture were smooth
	Gesture was comfortable
Time Consistency	Gesture timing was matched to speech
	Gesture speed were matched to speech
	Gesture pace was matched to speech
Semantics	Gesture was matched to speech content
	Gesture well described speech content
	Gesture helped me understand the content

We recruited 33 participants (17 male, 16 female, all native Japanese speakers, average = 38, standard deviation = 9.5 years old) through a cloud sourcing service. The results are shown in Fig. 3.

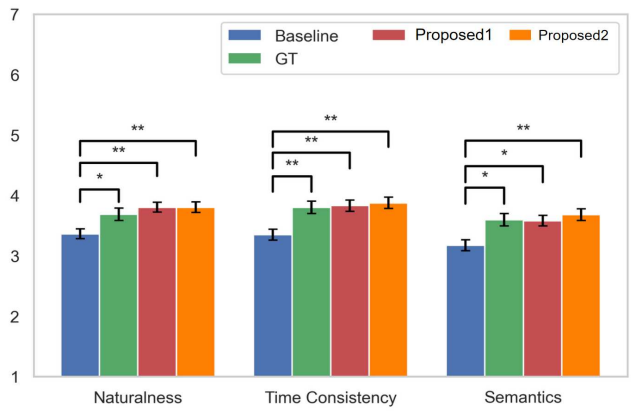

Figure 3: Subjective evaluation result. Proposed1 and proposed2 are the two synthesized results using the proposed model with different noise vector for the same audio segment input. *: $p < 0.05$, **: $p < 0.01$

Analysis of variance (ANOVA) was conducted to statistically test if there is a significant difference between the scores of the groups in our experiment setting. Scores of all scales passed ANOVA with $p < 0.01$. Tukey's honestly significant difference test (Tukey HSD) was used to test whether there is a significant difference between groups pair-wisely. For the naturalness scale, there was a significant difference between the baseline and ground truth , $p < 0.05$, between baseline and proposed 1 , $p < 0.01$, and between baseline and proposed 2 , $p < 0.01$. There was no significant difference between ground truth and proposed 1, $p = 0.77$, and between ground truth

and result 2, $p = 0.77$, and between proposed 1 and proposed 2, $p = 0.9$.

For the time consistency scale, there was a significant difference between baseline and ground truth , $p < 0.01$, and between baseline and proposed 1 , $p < 0.01$, and between baseline and proposed 2 , $p < 0.01$. There was no significant difference between ground truth and proposed 1, $p = 0.9$, between ground truth and result 2, $p = 0.9$, and between proposed 1 and proposed 2, $p = 0.9$.

For the semantics scale, there was a significant difference between baseline and ground truth , $p < 0.01$, between baseline and proposed 1 , $p < 0.05$, and between baseline and proposed 2 , $p < 0.01$. There was no significant difference between ground truth and proposed 1, $p = 0.9$, between ground truth and proposed 2, $p = 0.9$, and between proposed 1 and proposed 2, $p = 0.86$.

5 DISCUSSION AND LIMITATIONS

The results of the user study showed that the proposed model can not only yield better results than the baseline model, but also produce natural variations of motions. Additionally, there was no significant difference between the scores of the proposed model and the ground truth, suggesting that the naturalness of the gestures synthesized using the proposed model is approaching the ground truth level.

One main reason that the proposed model outperformed the baseline model is that the synthesized motion using the proposed model has a regular moving rate. On the other hand, the generated motion using the baseline model seems to move too much (i.e., once there are voices, it moves), which is not so natural. Although this can be partially attributed to the fact that the conversion network from coordinates to rotation vector is not optimal, mode collapse, which is a common failure in GAN training, seems to still be dominant for the baseline model. For example, the real data distribution has density on segments with movement and segments without movement for voiced input audio. Due to mode collapse, the baseline model only concentrates its density on part of the real distribution where the arms are moving frequently. As a result, the produced motions are moving too much compared with the real human motions. Instead, in the proposed method, we utilized an advanced training algorithm for WGAN, which reduces the effect of mode collapse, achieving a more human-like moving rate compared with the baseline model. In addition, motion holding (i.e., when the hands/arms are hold in the gesture space during some time) frequently appears in human gestures. Even though we did not explicitly model the frequency or appearance of motion holding in the produced gestures, we observed more motion holding than the baseline model from several synthesized samples in the proposed model. This could be another reason that the proposed model was better evaluated than the baseline model with respect to human perception.

On the other hand, no significant differences were found between the generated motions and the ground truth. However, it is hard to conclude that the generated motions have reached the human level. The reason is that the time of the video in the subjective evaluation is short (around 10 seconds). If the time of video for evaluation becomes longer, e.g. 1 minute, 10 minutes, or even 30 minutes, the evaluation results of the motions generated using the proposed model may be worse than the current result, since the model does not make efforts on learning any long-term dependencies. The effect of changing the length of the video used in the evaluation should be investigated in the future.

Also, although there should be connections between gestures and the context of the speech, these are not included in the current work. The reason is that high-dimensional conditions will drastically reduce the number of samples included in that condition, increasing the difficulty of training. In future work, however, the context in the speech should be included in the model to make the produced gestures more expressive.

6 CONCLUSION

In this paper, we proposed a GRU-based model for the generation of gestures from speech. A CNN-based discriminator was designed to train the generator with a WGAN-based learning algorithm in an adversarial manner, leading to better performance for the generator than a state-of-the-art baseline model. We investigated the effects of using a more stable learning algorithm for training GAN on gesture generation, and empirically provided a guideline for researchers who need a system of generating natural gestures from speech. More importantly, the achieved results can be directly applied to agents and robots whose movements are controlled by the joint rotation values, such as the Unity avatar used in this work.

A DETAILED MODEL ARCHITECTURE

The detailed model structure and hyper-parameters of the proposed model are provided in Fig. 4 and Fig. 5. The details of the Conv block in the discriminator are shown in Fig. 6. In the figures, FC represents fully-connected layer, and the numbers within brackets indicate input and output sizes of each block. For the 1D-Conv layers of the discriminator, the numbers within brackets indicate input channel, output channel, kernel size, and stride, respectively. For the LeakyReLU blocks, the negative slope is shown within the brackets.

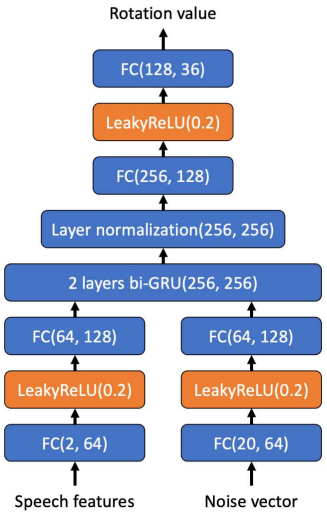

Figure 4: Generator architecture.

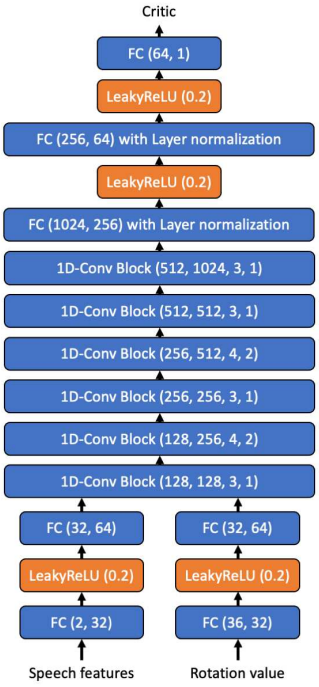

Figure 5: Discriminator architecture.

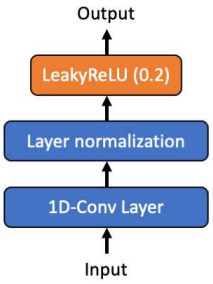

Figure 6: Structure of convolutional block used the discriminator.

REFERENCES

[1] 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*. Springer, 248–265.

[2] Chinenye Augustine Ajibo, Carlos Toshinori Ishi, Ryusuke Mikata, Chaoran Liu, and Hiroshi Ishiguro. 2020. Analysis of body gestures in anger expression and evaluation in android robot. *Advanced Robotics* 34, 24 (2020), 1581–1590.

[3] Simon Alexanderson, Gustav Eje Henter, Taras Kucherenko, and Jonas Beskow. 2020. Style-Controllable Speech-Driven Gesture Synthesis Using Normalising Flows. In *Computer Graphics Forum*, Vol. 39. Wiley Online Library, 487–496.

[4] Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein Generative Adversarial Networks. In *34th International Conference on Machine Learning*. 214–223.

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

[6] Ross Girshick. 2015. Fast r-cnn. In *IEEE International Conference on Computer Vision*. 1440–1448.

[7] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial networks. *arXiv preprint arXiv:1406.2661* (2014).

[8] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. [n.d.]. Improved Training of Wasserstein GANs. In *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.). Curran Associates, Inc.

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

[10] Carlos Toshinori Ishi, Hiroshi Ishiguro, and Norihiro Hagita. 2008. Automatic extraction of paralinguistic information using prosodic features related to F0, duration and voice quality. *Speech communication* 50, 6 (2008), 531–543.

[11] Carlos T Ishi, Daichi Machiyashiki, Ryusuke Mikata, and Hiroshi Ishiguro. 2018. A speech-driven hand gesture generation method and evaluation in android robots. *IEEE Robotics and Automation Letters* 3, 4 (2018), 3757–3764.

[12] Ryo Ishii, Taichi Katayama, Ryuichiro Higashinaka, and Junji Tomita. 2018. Generating body motions using spoken language in dialogue. In *18th International Conference on Intelligent Virtual Agents*. 87–92.

[13] Durk P Kingma and Prafulla Dhariwal. [n.d.]. Glow: Generative Flow with Invertible 1x1 Convolutions. In *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.). Curran Associates, Inc.

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

[15] Taras Kucherenko, Dai Hasegawa, Naoshi Kaneko, Gustav Eje Henter, and Hedvig Kjellström. 2021. Moving fast and slow: Analysis of representations and post-processing in speech-driven automatic gesture generation. *International Journal of Human–Computer Interaction* (2021), 1–17.

[16] 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 *International Conference on Multimodal Interaction*. 242–250.

[17] Yukiko I Nakano, Masashi Okamoto, Daisuke Kawahara, Qing Li, and Toyoaki Nishida. 2004. Converting text into agent animations: Assigning gestures to text. In *HLT-NAACL 2004: Short Papers*. 153–156.

[18] Manan Oza, Himanshu Vaghela, and Kriti Srivastava. 2019. Progressive Generative Adversarial Binary Networks for Music Generation. *arXiv:1903.04722* (2019).

[19] Sai Rajeswar, Sandeep Subramanian, Francis Dutil, Christopher Pal, and Aaron Courville. 2017. Adversarial Generation of Natural Language. *arXiv:1705.10929* (2017).

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

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

[22] Bowen Wu, Chaoran Liu, Carlos Toshinori Ishi, and Hiroshi Ishiguro. 2021. Modeling the Conditional Distribution of Co-Speech Upper Body Gesture Jointly Using Conditional-GAN and Unrolled-GAN. *Electronics* 10, 3 (2021), 228.

[23] 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 (TOG)* 39, 6 (2020), 1–16.

[24] Chuang Yu and Adriana Tapus. 2020. SRG 3: Speech-driven Robot Gesture Generation with GAN. In *16th International Conference on Control, Automation, Robotics and Vision (ICARCV)*. IEEE, 759–766.

[25] Yi Zhao, Shinji Takaki, Hieu-Thi Luong, Junichi Yamagishi, Daisuke Saito, and Nobuaki Minematsu. 2018. Wasserstein GAN and Waveform Loss-based Acoustic Model Training for Multi-speaker Text-to-Speech Synthesis Systems Using a WaveNet Vocoder. arXiv:1807.11679 [eess.AS]