HUMANGAN: GENERATIVE ADVERSARIAL NETWORK WITH HUMAN-BASED DISCRIMINATOR AND ITS EVALUATION IN SPEECH PERCEPTION MODELING

Kazuki Fujii[1,2], *Yuki Saito*[2], *Shinnosuke Takamichi*[2], *Yukino Baba*[3], *and Hiroshi Saruwatari*[2]

[1] National Institute of Technology, Tokuyama College, Japan.
[2] Graduate School of Information Science and Technology, The University of Tokyo, Japan.
[3] Faculty of Engineering, Information and Systems, University of Tsukuba, Japan.

ABSTRACT

We propose the HumanGAN, a generative adversarial network (GAN) incorporating human perception as a discriminator. A basic GAN trains a generator to represent a real-data distribution by fooling the discriminator that distinguishes real and generated data. Therefore, the basic GAN cannot represent the outside of a real-data distribution. In the case of speech perception, humans can recognize not only human voices but also processed (i.e., a non-existent human) voices as human voice. Such a human-acceptable distribution is typically wider than a real-data one and cannot be modeled by the basic GAN. To model the human-acceptable distribution, we formulate a backpropagation-based generator training algorithm by regarding human perception as a black-boxed discriminator. The training efficiently iterates generator training by using a computer and discrimination by human. We evaluate our HumanGAN in speech naturalness modeling and demonstrate that it can represent a human-acceptable distribution that is wider than a real-data distribution.

Index Terms— generative adversarial network, human computation, black-box optimization, crowdsourcing, speech perception

1. INTRODUCTION

Generative models in machine learning have contributed strongly to media (e.g., speech)-related research. In particular, deep neural network (DNN)-based generative models, which are called “deep generative models,” can model a complicated data distribution thanks to the nonlinear transformation of DNNs. A generative adversarial network (GAN) [1] is one of the most promising approaches in learning deep generative models. The basic GAN framework includes not only a generator but also a discriminator. The basic GAN trains the discriminator by distinguishing real and generated data and separately trains the generator by fooling the discriminator. By iterating the training steps, the generator can randomly generate samples that follow a real-data distribution, i.e., training-data distribution. Many studies on speech [2], image [3], and language [4] have successfully applied the basic GAN framework, and more studies are expected to appear in speech-related research.

However, the basic GAN represents only a real-data distribution[1]. Human speech perception can accept a deviation from real speech. For instance, humans can recognize both human speech and processed (i.e., non-existent human) speech as human voice. In this paper, we call such a human-acceptable distribution the *perception distribution* and establish a GAN framework that can represent it. If the perception distribution covers wider ranges than a real-data distribution, a basic GAN trained by fooling the discriminator could never represent the perception distribution. Also, collecting a large amount of real data for training the basic GAN would never solve this problem.

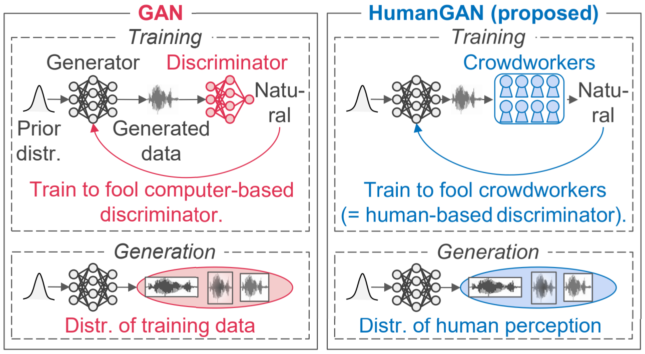

Fig. 1. Comparison of basic GAN and proposed HumanGAN. Basic GAN trains generator by fooling DNN-based discriminator (i.e., computer-based discriminator), and generator finally represents training-data distribution. In comparison, HumanGAN trains generator by fooling crowdworkers’ perceptual evaluation (i.e., human-based discriminator), and generator finally represents humans’ perception distribution.

In this paper, we propose the *HumanGAN* trained by fooling human perception to model a human perception distribution. The HumanGAN regards a crowd as a black-boxed system that outputs a difference of posterior probabilities (i.e., to what degree do humans accept the data) given generated data. We formulate a backpropagation-based generator training algorithm utilizing an optimization algorithm for the black-boxed system. **Figure 1** shows a comparison of the basic GAN and proposed HumanGAN. Whereas a generator of the basic GAN fools a DNN-based discriminator, that of the proposed HumanGAN fools perceptual evaluation performed by the crowdworkers of a crowdsourcing service. In this paper, we evaluate the HumanGAN in the modeling of speech naturalness, i.e., to what degree can humans accept synthesized speech as human voice. The experimental results demonstrate that 1) the perception distribution covers a wider range than the real-data distribution, and 2) the proposed HumanGAN accurately models the perception distribution, which the basic GAN cannot.

2. BASIC GAN USING DNN-BASED DISCRIMINATOR

The aim of the basic GAN is to match a generated-data distribution and real-data distribution. To achieve this, the basic GAN

[1] This problem is common among generative models, but this paper deals only with GAN.

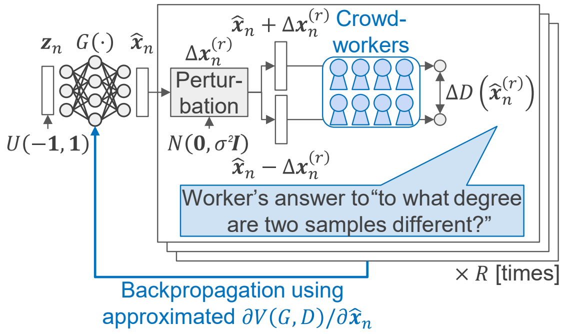
z_n
$G(\cdot)$
$\hat{x}_n$
$U(-1,1)$
$\Delta x_n^{(r)}$
Pertur-bation
$N(0,\sigma^2 I)$
$\hat{x}_n + \Delta x_n^{(r)}$
Crowd-workers
$\hat{x}_n - \Delta x_n^{(r)}$
$\Delta D\left(\hat{x}_n^{(r)}\right)$
Worker's answer to"to what degree are two samples different?"
× R [times]
Backpropagation using approximated $\partial V(G,D)/\partial\hat{x}_n$

Fig. 2. Generator training procedure of proposed HumanGAN. Crowdworkers state a perceptual difference (i.e., difference of posterior probabilities) of two perturbed samples. Answer and perturbation are used for backpropagation to train generator.

uses not only a generator $G(\cdot)$ that randomly generates data but also a discriminator $D(\cdot)$ that distinguishes real and generated data. Here, let real data be $\boldsymbol{x} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n, \cdots, \boldsymbol{x}_N]$, and N be the number of data. The generator $G(\cdot)$ transforms prior noise $\boldsymbol{z} = [\boldsymbol{z}_1, \cdots, \boldsymbol{z}_n, \cdots, \boldsymbol{z}_N]$ into generated data $\hat{\boldsymbol{x}} = [\hat{\boldsymbol{x}}_1, \cdots, \hat{\boldsymbol{x}}_n, \cdots, \hat{\boldsymbol{x}}_N]$. The prior noise follows a known probability distribution, e.g., a uniform distribution. The input of the discriminator $D(\cdot)$ is real data $\boldsymbol{x}_n$ or generated data $\hat{\boldsymbol{x}}_n$, and the $D(\cdot)$ outputs a posterior probability that the input is real data. An objective function in training is formulated as

$$V(G, D) = \sum_{n=1}^{N} \log D(\boldsymbol{x}_n) + \sum_{n=1}^{N} \log\left(1 - D(G(\boldsymbol{z}_n))\right). \quad (1)$$

The generator $G(\cdot)$ and discriminator $D(\cdot)$ are trained one by one. The following sections describe these training steps.

2.1. Generator training

The generator training step minimizes Eq. (1) to estimate the model parameters of $G(\cdot)$. The model parameter θ_{G} is estimated:

$$\theta_{\mathrm{G}} = \underset{\theta_{\mathrm{G}}}{\operatorname{argmin}} \sum_{n=1}^{N} \log\left(1 - D(G(\boldsymbol{z}_n))\right). \quad (2)$$

Namely, $G(\cdot)$ is trained to let $D(\cdot)$ recognize the generated data as "real." In general, all computation processes are differentiable, and θ_{G} is iteratively estimated by using the backpropagation algorithm.

2.2. Discriminator training

The discriminator training step maximizes Eq. (1) to estimate the model parameters of $D(\cdot)$. The model parameter θ_{D} is estimated:

$$\theta_{\mathrm{D}} = \underset{\theta_{\mathrm{D}}}{\operatorname{argmax}}\, V(G, D). \quad (3)$$

2.3. Problem

When the generator is accurately trained, it can randomly sample data that follows a real-data distribution. However, when the perception distribution covers ranges wider than the real-data distribution, the basic GAN cannot represent the ranges. We show such an actual example in speech perception in **Section 4.2**.

3. HUMANGAN USING HUMAN-BASED DISCRIMINATOR

We propose the HumanGAN to represent humans' perception distribution. As shown in **Fig. 1**, the HumanGAN replaces the DNN-based discriminator of the basic GAN with humans' perceptual evaluation. The use of a DNN-based generator and prior noise are common between the basic GAN and HumanGAN. However, whereas the basic GAN trains the generator by fooling the DNN-based discriminator, the HumanGAN trains one by fooling humans' perceptual evaluation. Here, we re-define $D(\cdot)$ as a perception model that represents humans' perceptual evaluation. An input of the $D(\cdot)$ is $\hat{\boldsymbol{x}}_n$ generated from $G(\cdot)$, and the $D(\cdot)$ outputs a posterior probability about "to what degree is the input perceptually acceptable" from 0 to 1. The objective function during training is as follows.

$$V(G, D) = \sum_{n=1}^{N} D(G(\boldsymbol{z}_n)). \quad (4)$$

As described in this equation, the HumanGAN never uses real-data in training.

3.1. Generator training

A model parameter θ_{G} of $G(\cdot)$ is estimated by maximizing Eq. (4). In this paper, we formulate a gradient-based iterative method. Namely, θ_{G} is iteratively updated as follows.

$$\theta_{\mathrm{G}} \leftarrow \theta_{\mathrm{G}} + \alpha \frac{\partial V(G, D)}{\partial \theta_{\mathrm{G}}}, \quad (5)$$

where α is the learning coefficient. The second term $\partial V(G, D)/\partial\theta_{\mathrm{G}}$ is decomposed into a product of $\partial V(G, D)/\partial\hat{\boldsymbol{x}}$ and $\partial\hat{\boldsymbol{x}}/\partial\theta_{\mathrm{G}}$. In the basic GAN, $\partial V(G, D)/\partial\theta_{\mathrm{G}}$ is estimated by using the standard backpropagation algorithm since the computation processes of both $G(\cdot)$ and $D(\cdot)$ are differentiable. However, $D(\cdot)$ is not differentiable in the HumanGAN. Therefore, we cannot estimate $\partial V(G, D)/\partial\hat{\boldsymbol{x}}$ and cannot use the standard backpropagation algorithm. In this paper, we regard humans as a black-boxed system that outputs a difference of the posterior probabilities of generated data and estimate $\partial V(G, D)/\partial\hat{\boldsymbol{x}}$ on the basis of an optimization algorithm for black-boxed systems. We propose a training algorithm that uses natural evolution strategies (NES) [5] that approximates gradients by using data perturbations. **Figure 2** shows the training procedure.

First, a small perturbation $\Delta\boldsymbol{x}_n^{(r)}$ is randomly generated from a multivariate Gaussian distribution $\mathcal{N}\left(\boldsymbol{0}, \sigma^2\boldsymbol{I}\right)$. r is the perturbation index ($1 \leq r \leq R$). σ is a constant value of standard deviation. $\boldsymbol{I}$ is the identity matrix. Then, a human is presented with two perturbed data $\left\{\hat{\boldsymbol{x}}_n + \Delta\boldsymbol{x}_n^{(r)}, \hat{\boldsymbol{x}}_n - \Delta\boldsymbol{x}_n^{(r)}\right\}$ and evaluates the difference of their posterior probabilities:

$$\Delta D(\hat{\boldsymbol{x}}_n^{(r)}) \equiv D\left(\hat{\boldsymbol{x}}_n + \Delta\boldsymbol{x}_n^{(r)}\right) - D\left(\hat{\boldsymbol{x}}_n - \Delta\boldsymbol{x}_n^{(r)}\right). \quad (6)$$

$\Delta D(\hat{\boldsymbol{x}}_n^{(r)})$ ranges from -1 to 1. For example, we expect that a human will return $\Delta D(\hat{\boldsymbol{x}}_n^{(r)}) = 1$ when he/she perceives that $D(\hat{\boldsymbol{x}}_n + \Delta\boldsymbol{x}_n^{(r)})$ is greatly higher than $D(\hat{\boldsymbol{x}}_n - \Delta\boldsymbol{x}_n^{(r)})$. This perturbation and evaluation are iterated R times for one $\hat{\boldsymbol{x}}_n$. Finally, these processes are iterated for N generated data.

$\partial V(G, D)/\partial\hat{\boldsymbol{x}}$ for the backpropagation algorithm is approximated as [5]

$$\frac{\partial V(G, D)}{\partial\hat{\boldsymbol{x}}} = \left[\frac{\partial V(G, D)}{\partial\hat{\boldsymbol{x}}_1}, \cdots, \frac{\partial V(G, D)}{\partial\hat{\boldsymbol{x}}_n}, \cdots, \frac{\partial V(G, D)}{\partial\hat{\boldsymbol{x}}_N}\right], \quad (7)$$

$$\frac{\partial V(G, D)}{\partial\hat{\boldsymbol{x}}_n} = \frac{1}{2\sigma^2 R} \sum_{r=1}^{R} \Delta D(\hat{\boldsymbol{x}}_n^{(r)}) \cdot \Delta\boldsymbol{x}_n^{(r)}. \quad (8)$$

Here, we summarize the number of queries to humans for training the generator. Since the perturbation is performed R times for each generated data, the total number of queries is a product of the number of training iterations, the number of generated data N, and the number of perturbations R.

3.2. Discussion

Our work utilizes the idea of *human computation* [6] in which a machine performs its function by outsourcing inner steps to humans, and the HumanGAN is regarded as a human-in-the-loop machine learning technique. Some existing studies made humans have a function and integrated it into DNN training. Sakata et al. [7] regarded humans as a feature extractor and improved the performance of a multi-class classifier. Also, Jaques et al. [8] regarded humans as an auxiliary classifier and improved performances of a generator. From these viewpoints, our work regards humans as a system that outputs a difference of posterior probabilities and represents humans' perceptual distribution.

Koyama et al.'s work [9] is related work from the viewpoint of exploring humans' perception. However, their aim is completely different from ours. Koyama et al. made humans score a perceptual difference of two data and aimed at visualizing the perception distribution by interpolating the scores. In comparison, our work makes humans score the perceptual difference of a perturbation and aims at training a generator that represents the perception distribution.

Natural language-based image generation and editing [10, 11] are related work from the viewpoint of modeling a data distribution that corresponds to human intention. However, these methods only estimate a conditional distribution within a real-data distribution. In comparison, our work estimates a perception distribution even if it is wider than the real-data distribution.

The basic GAN utilizes only a computer-based discriminator, and the HumanGAN utilizes only a human-based discriminator. One part of our future work involves the use of a human-computer discriminator, i.e., the generator is trained by using both computer-based and human-based discriminators. Related studies have proposed active learning algorithms that use the basic GAN [12, 13]. These studies are currently limited to real-data distribution modeling but have the potential to be applied to perception distribution modeling.

Data imbalance in training data is a general problem in machine learning. Since the HumanGAN does not require real data for the training, it has the possibility of relaxing the problem.

3.3. Practical limitation

Here, we list empirically found limitations of the HumanGAN.

As described in Eq. (4), the generator of the HumanGAN is trained by maximizing the posterior probability. Therefore, a mode collapse problem [14] is caused when data generated from the initial generator is distributed near the mode of the perception distribution. Also, a gradient vanishing problem is caused when generated data is distributed far from the perception distribution. In the evaluation in **Section 4.3**, we used real data for initializing the generator.

The setting of σ in the NES algorithm changes the degree of perturbation. When σ is very small, the discriminator of the HumanGAN, i.e., crowdworkers, cannot find a perceptual difference, and it makes gradients vanish. When σ is very large, the gradient becomes inaccurate. In the preliminary experiments of the following evaluation, we used several settings for σ and finally determined one value.

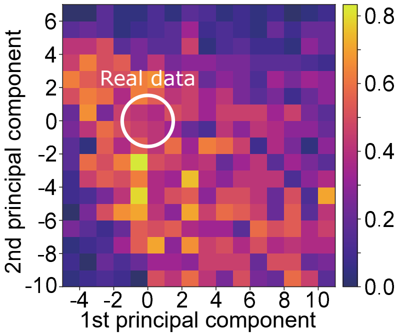

Fig. 3. Color map representing posterior probabilities in speech naturalness.

As described in **Section 3.1**, the number of queries to crowdworkers is a product of the number of training iterations, the number of data N, and the number of perturbations R. The number of queries increases the monetary and time costs for training. In the following evaluation, we used a small generator, a small number of generated data, and low-dimensional data to limit the costs.

4. EXPERIMENTAL EVALUATION

4.1. Experimental setup

An experimental evaluation was done to investigate whether our HumanGAN can represent the perception distribution of speech naturalness, i.e., the HumanGAN tries to represent a distribution of speech features for which humans recognize the synthesized speech as human speech. Although the HumanGAN originally does not use real data, this evaluation used speech data for comparison with the basic GAN, dimensional reduction of speech features, and initialization of the generator. The speech data used was reading-style speech uttered by 199 female speakers included in the Vowel Database: Five Japanese Vowels of Males, Females, and Children Along with Relevant Physical Data (JVPD) [15]. Before extracting speech features, the speech data were downsampled at 16 kHz, and their powers were normalized. The speech features were 513-dimensional log spectral envelopes extracted every 5 ms. The WORLD vocoder [16, 17] was used for the feature extraction. Julius [18] was adopted to obtain phoneme alignment, and we extracted speech features of the Japanese vowel /a/. We applied principal component analysis (PCA) to the log spectral envelopes of /a/ of 199 female speakers and extracted two-dimensional principal components. In this paper, we assumed that the real data of many speakers follows a two-dimensional standard Gaussian distribution, and we normalized the two-dimensional principal components to have zero-mean and unit-variance. For humans' perceptual evaluation, a speech waveform was synthesized from the generated speech features. First, the first and second principal components were generated from a generator and de-normalized. The remaining speech features, i.e., the third and upper principal components, the fundamental frequency, and aperiodicity components, were copied from one frame of an average-voice speaker. Average-voice speaker means a speaker whose first and second principal components are the closest to 0. These correspond to speech features of one frame (5 ms). To make the perceptual evaluation easy for crowdworkers,we copied the features for 200 frames and synthesized 1 s of a speech waveform by using the WORLD vocoder.

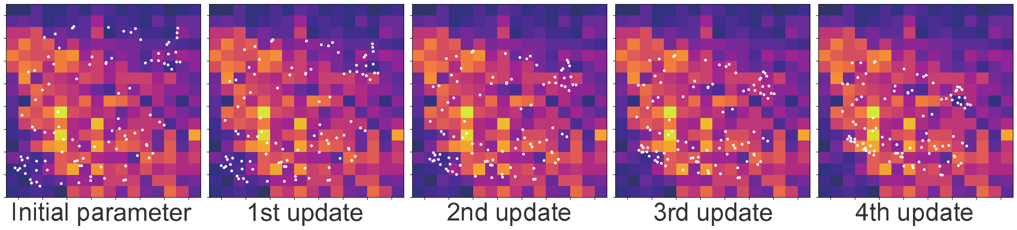

Fig. 4. Generated data (white point) at each training iteration. Color map shown in **Fig. 3** was also drawn but never used in training.

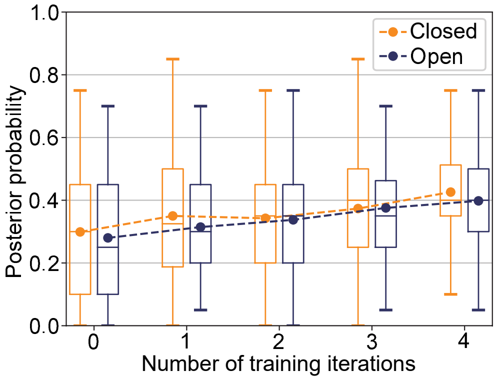

Fig. 5. Posterior probability at each iteration step. Box indicates first, second (i.e., median), and third quartiles. Line plot indicates mean value.

4.2. Difference between real-data and perception distributions

First, to emphasize the need for the HumanGAN, we demonstrate that a perception distribution is different from a real-data distribution in speech naturalness. We split the two-dimensional space into grids and evaluated humans' naturalness tolerance (i.e., posterior probability of speech naturalness). For the humans' evaluation, we instructed listeners to "listen to a voice and answer with 1 when perceiving the speech as very unnatural and 5 when perceiving it as very natural" and had listeners give a score on a 5-point scale (from 1 to 5). 1-through-5 of the obtained scores corresponded to $0.00, 0.25, 0.50, 0.75, 1.00$ of the posterior probability $D\left(\hat{\boldsymbol{x}}_n\right)$. The evaluation was performed by using the Lancers crowdsourcing platform [19]. The posterior probability of each grid was set to the value averaged among listeners. At least five listeners scored for one grid, and the total number of listeners was 96.

Figure 3 shows the result. As described in **Section 4.1**, the real data was normalized to have zero-mean and unit-variance, and the basic GAN represents this range. However, as shown in **Fig. 3**, the perception distribution covered a range wider than the real-data distribution. The basic GAN cannot represent this wider range, so the HumanGAN, which adopts perceptual evaluation by humans is needed for modeling the perception distribution.

4.3. Transition of generated data during training

Next, to intuitively understand generator training, we plotted the data generated at each iteration step. The prior noise fed to the generator followed a two-dimensional uniform distribution $U(-1,1)$. This prior noise was randomly generated before a training iteration and fixed during the iteration. The generator was a small feed-forward neural network consisting of a two-unit input layer, 2×4-unit sigmoid hidden layers, and two-unit linear output layer. The model parameters were randomly initialized, but we iterated the random initialization until the initial generator output data that covered ranges of a higher posterior probability, shown in **Fig. 3**. The gradient descent method with a learning rate of $\alpha = 0.0015$ was used for training. Chainer [20] was used for implementation. The number of generated data N, the number of perturbations, and the number of training iterations were set to 100, 5, and 4, respectively. The standard deviation of NES was set to 1.0. During the HumanGAN training, we instructed listeners to "listen to two voices and answer with 1 when perceiving the first one as significantly natural and 5 when perceiving the second as significantly natural. Answer 3 when perceiving the two samples as equally natural." The 1-through-5 of the obtained scores corresponded to $1.0, 0.5, 0.0, -0.5, -1.0$ of the difference of the posterior probabilities, $\Delta D(\hat{\boldsymbol{x}}_n^{(r)})$.

Figure 4 shows the data generated for each iteration step. The posterior probability of **Fig. 3** is also drawn, but note that we never used it in training. The figure indicates that the generated data transited from a lower probability range (darker area) to a higher probability range (brighter area) during the training iteration. Therefore, we can qualitatively demonstrate that the HumanGAN makes it possible to train a generator that represents a human's perception distribution.

4.4. Increase of posterior probability in training

Finally, we confirmed that the training iteration increases the posterior probability. Here, we sampled new prior noise not used in training and generated new speech features. We call speech features generated from prior noise used in training "closed data" and those generated from prior noise not used in training "open data." The posterior probabilities of the open data were scored by humans in the same manner as in **Section 4.2**.

Figure 5 shows a box plot at each iteration step. We can see that the training iteration increases the posterior probabilities of not only the closed data but also the open data. Therefore, we can quantitatively demonstrate that the generator of the HumanGAN can represent the perception distribution.

5. CONCLUSION

This paper presented the HumanGAN, which can represent humans' perceptually recognizable distribution. The DNN-based discriminator of the basic GAN was replaced with human-based perceptual evaluation. The DNN-based generator of the HumanGAN was trained by using a black-boxed optimization algorithm and human evaluation. We evaluated the HumanGAN in speech naturalness modeling, and we qualitatively and quantitatively demonstrated that the HumanGAN can represent a humans' perceptually recognizable distribution. As future work, we address the scalability of the HumanGAN in terms of the number of data and feature dimensions.

Acknowledgements: This research and development was supported by the SECOM Science and Technology Foundation and the MIC/SCOPE #182103104.

6. REFERENCES

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. NIPS*, Montreal, Canada, Dec. 2014, pp. 2672–2680.

[2] Y. Saito, S. Takamichi, and H. Saruwatari, "Statistical parametric speech synthesis incorporating generative adversarial networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 1, pp. 84–96, Jan. 2018.

[3] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv:1411.1784*, 2015.

[4] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu, "SeqGAN: Sequence generative adversarial nets with policy gradient," in *Proc. AAAI*, San Francisco, U.S.A., Feb. 2017.

[5] A. Ilyas, L. Engstrom, A. Athalye, and J. Lin, "Black-box adversarial attacks with limited queries and information," in *Proc. ICML*, Stockholm, Sweden, Jul. 2018, vol. 2, pp. 2137–2146.

[6] A. J Quinn and BB Bederson, "Human computation: a survey and taxonomy of a growing field," in *CHI '11 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Vancouver, Canada, May 2011, pp. 1403–1402.

[7] Y. Sakata, Y. Baba, and H. Kashima, "CrowNN: Human-in-the-loop network with crowd-generated inputs," in *Proc. ICASSP*, Brighton, U.K., May 2019, pp. 7555–7559.

[8] N. Jaques, J. McCleary, J. Engel, D. Ha, F. Bertsch, R. Picard, and D. Eck, "Learning via social awareness: Improving a deep generative sketching model with facial feedback," in *arXiv preprint arXiv:1802.04877*, 2018.

[9] Y. Koyama, D. Sakamoto, and T. Igarashi, "Crowd-powered parameter analysis for visual design exploration," in *Proc. UIST*, New York, U.S.A., Oct. 2014, pp. 65–74.

[10] E. Mansimov, E. Parisotto, J. Ba, and R. Salakhutdinov, "Generating images from captions with attention," in *Proc. ICLR*, San Juan, Puerto Rico, May 2016.

[11] S. Shinagawa, K. Yoshino, S. Sakriani, Y. Suzuki, and S. Nakamura, "Image manipulation system with natural language instruction," *IEICE Transactions on Information and Systems*, vol. J102-D, no. 8, pp. 514–529, Aug. 2019 (in Japanese).

[12] J.-J. Zhu and J. Bento, "Generative adversarial active learning," in *arXiv preprint arXiv:1702.07956*, 2017.

[13] Y. Deng, K. Chen, Y. Shen, and H. Jin, "Adversarial active learning for sequences labeling and generation," in *Proc. IJCAI*, Stockholm, Sweden, Jul. 2018, pp. 4012–4018.

[14] I. Goodfellow, "NIPS 2016 tutorial: Generative adversarial networks," in *Proc. NIPS*, Barcelona, Spain, Dec. 2016.

[15] "Vowel database: Five Japanese vowels of males, females, and children along with relevant physical data (JVPD)," `http://research.nii.ac.jp/src/en/JVPD.html`.

[16] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: a vocoder-based high-quality speech synthesis system for real-time applications," *IEICE transactions on information and systems*, vol. E99-D, no. 7, pp. 1877–1884, Jul. 2016.

[17] M. Morise, "D4C, a band-aperiodicity estimator for high-quality speech synthesis," *Speech Communication*, vol. 84, pp. 57–65, Nov. 2016.

[18] A. Lee, T. Kawahara, and K. Shikano, "Julius — an open source real-time large vocabulary recognition engine," in *Proc. EUROSPEECH*, Aalborg, Denmark, Sep. 2001, pp. 1691–1694.

[19] "Lancers," `https://www.lancers.jp/`.

[20] S. Tokui, R. Okuta, T. Akiba, Y. Niitani, T. Ogawa, S. Saito, S. Suzuki, K. Uenishi, B. Vogel, and H. Y Vincent, "Chainer: A deep learning framework for accelerating the research cycle," in *KDD '19 Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Anchorage, AK, U.S.A., Aug. 2019, pp. 2002–2011.