Amphibian Sounds Generating Network Based on Adversarial Learning

Sangwook Park, Mounya Elhilali, *Senior Member, IEEE*, David K. Han, *Senior Member, IEEE*, and Hanseok Ko, *Senior Member, IEEE*

***Abstract*—This letter proposes a generative network based on adversarial learning for synthesizing short-time audio streams and investigates the effectiveness of data augmentation for amphibian call sounds classification. Based on Fourier analysis, the generator is designed by a multi-layer perceptron composed of frequency basis learning layers and an output layer, and a discriminator is constructed by a convolutional neural network. Additionally, regularization on weights is introduced to train the networks with practical data that includes some disturbances. Synthetic audio streams are evaluated by quantitative comparison using inception score, and classification results are compared for real versus synthetic data. In conclusion, the proposed generative network is shown to produce realistic sounds and therefore useful for data augmentation.**

***Index Terms*—Generative model, adversarial networks, Wasserstein distance, audio stream generation.**

I. Introduction

ACCESS to data (e.g., speech corpora, video datasets) has led to great advances in signal processing and effective deployment of deep learning techniques for recognition, discrimination and detection tasks. Still, progress in a number of niche domains remains limited due to lack of suitable data; one such domain is natural conservation and presentation of biodiversity of habitats. Around the world, authorities engage in tedious efforts of observing fauna populations in given regions in order to assess fluctuation in species over a period of time [1]–[3]. With technological advances, automatic monitoring systems are being explored where experts use sound recognition technologies to track different species. Amphibian population tracking is one area where automated systems based on sound recordings have been explored by collecting sounds from various frog and toad species [3]–[6].

As data collection is an expensive and intricate effort, the idea of using generative networks to augment data has begun to attract the attention of researchers [7]–[9]. Among several approaches, Generative Adversarial Networks (GANs) composed of generator and discriminator have been remarkably effective in image generation [10]–[14]. The generator produces an image from a random seed, and the discriminator decides whether the image was synthesized by the generator or not. With help of regularization schemes and tuning [15]–[18], the training of GAN may converge, and the generator produces data indistinguishable by the discriminator.

Although GAN-based data generation has been effective for image-related tasks, using GAN networks for audio generation still poses a number of challenges. Generating a raw audio streams requires an elaborate model for estimating samples that are composed of a raw audio stream. If some of the samples in a synthetic stream are under- or over-estimated, the synthesized audios are different to targets that a generator is intended to produce. Mun, *et al.* [19] avoided this difficulty by producing audio features that are robust to displacements rather than raw audio streams. As an alternative to GANs, WaveNet is a different representative method for raw audio stream generation [20]. WaveNet uses a large receptive field with dilated convolutions that make up an auto-regression model for estimating downstream samples based on previous ones. Motivated by the auto-regression model of WaveNet, Donahue *et al.* [21] proposed an audio-stream generating network using GAN, named as “WaveGAN”, which was designed by applying 1D transposed convolution and a large up-sampling factor. One of limitations is the use of large-size receptive fields, which make such networks unsuitable for generating short audio streams like amphibian call sounds. The mismatch in size between desired short audio target and the large response fields in the network causes noise to fill the size gap, which in turn results in generation of noise streams.

To handle short-time audio generation, this letter proposes a generative network that constrains the signal structure by its frequency basis. The generator is designed as a Multi-Layer Perceptron (MLP) composed of frequency basis learning layers and an output layer. A Convolutional Neural Network (CNN) is employed as discriminator. The use of realistic data recorded in real environments helps constrain the regularization of the network. The effectiveness of this network is demonstrated by performing amphibian call sounds generation. The contributions of this letter are as follows: 1) development of a GAN based generative model for synthesizing short-time audio streams; and 2) investigation of feasibility of data augmentation with synthesized audio clips by the proposed network.

Manuscript received February 23, 2020; accepted April 10, 2020. Date of publication April 20, 2020; date of current version May 12, 2020. This work was supported in part by the Korea Environment Industry & Technology Institute (KEITI) through the Public Technology Program based on Environmental Policy and in part by the Korea Ministry of Environment (MOE) under Grant 2017000210001. The work of David K. Han was supported by the U.S. Army Research Laboratory. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Daniele Giacobello. *(Corresponding author: Hanseok Ko.)*

Sangwook Park and Mounya Elhilali are with the Department of Electrical and Computer Engineering, Johns Hopkins University, Baltimore, MD 21218 USA (e-mail: spark190@jhu.edu; mounya@jhu.edu).

David K. Han is with the U.S. Army Research Laboratory, Adelphi, MD 20783 USA (e-mail: ctmkhan@gmail.com).

Hanseok Ko is with the School of Electrical Engineering, Korea University, Seoul 02841, Republic of Korea (e-mail: hsko@korea.ac.kr).

This letter has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors.

Digital Object Identifier 10.1109/LSP.2020.2988199

II. Related Works

Goodfellow *et al.* [10] proposed an adversarial modeling of a generator $G(z;\theta_G)$ and a discriminator $D(x;\theta_D)$. The generator *G* maps a random vector *z* into a data space while the discriminator *D* represents the probability that *x* came from the real data space. The objective function for GAN is denoted as $f_{GAN}(\theta_G,\theta_D) = E_{Pdata}(x)[log(D(x;\theta_D)] + E_{Pz(z)}[log(1 - D(G(z;\theta_G);\theta_D)]$ where $E_{Pdata}[.]$ and $E_{Pz(z)}$ [.] means an expectation operator to probability density function of real data and random vector, respectively. The parameters of the generator θ_G are trained to minimize the function while the discriminator θ_D attempts to maximize the function. In practice, θ_G and θ_D are alternatively and repetitively trained with the other held fixed [10],

$$\theta_D^n = \arg\max_{\theta_D} f_{GAN}(\theta_G^{n-1},\theta_D)$$

$$\theta_G^n = \arg\min_{\theta_G} f_{GAN}(\theta_G,\theta_D^n), \tag{1}$$

where *n* is the number of iterations.

A number of studies modified the objective function. Mao *et al.* [22] proposed Least Squares GAN (LSGAN) that adopted the least square error as $f_{LSGAN}(\theta_G;\theta_D) = E_{P\mathrm{data(x)}}[(D(x;\theta_D) - 1)^2] + E_{P\mathrm{z(z)}}[(D(G(z;\theta_G);\theta_D))^2]$. The study reported that the least square loss forces the synthetic data toward the decision boundary resulting in synthetic data being in close proximity to real data.

Arjovsky *et al.* [23] proposed *Wasserstein GAN* (WGAN) for stable learning. In WGAN, Earth-Mover (EM), i.e., Wasserstein distance is used for an objective function instead of alternatives such as Kullback-Leibler (KL) divergence and Jensen-Shannon (JS) divergence which are well known as a distance measure between distributions [24]. According to this study, KL or JS divergence is considered inappropriate as an objective function due to its discontinuities. On the other hand, the Wasserstein distance is continuous and differentiable in most of the parameter domain if the discriminator satisfies the Lipschitz condition. The objective function for WGAN is defined as $f_{\mathrm{WGAN}}(\theta_G;\theta_D) = E_{P\mathrm{data(x)}}[D(x;\theta_D)] - E_{P\mathrm{z(z)}}[D(G(z;\theta_G);\theta_D)]$. Both θ_G and θ_D are updated toward maximizing the function. In order to satisfy the Lipschitz condition, θ_D has to be clipped on the interval [$-c$, *c*]. Since it is hard to optimize the parameter *c*, however, Petzka, *et al.* [16] proposed a modified gradient penalty to satisfy the condition as

$$f_{WGAN\_LP} = f_{WGAN} - \lambda_{LP} E_{\hat{x}}[(\max\{0, \|\nabla_{\hat{x}} D(\hat{x})\| - 1\})^2]. \tag{2}$$

where $\hat{x}$ is an internally dividing data between the real and synthetic data from the generator.

III. Proposed Method

A. Signal Model With Respect to Neural Network

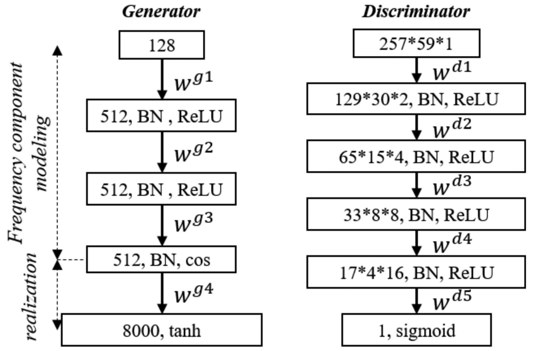

Fig. 1. Network architectures of generator and discriminator for signal generation; the weight, w^{g4}, for realization satisfies the weight condition as $w^{g4}(i, j) = w^{g4}(i + 256, j)$, for $1 \le i \le 256$, $1 \le j \le 8000$.

In the Fourier domain, an arbitrary signal can be represented as a linear combination of frequency bases. Therefore, a single frame of an audio recording, x_n, can be modeled as a signal component s_n (represented by its frequency elements) as well as a noise component d_n, added together as

$$x_n = \frac{1}{K}\sum_{k=0}^{K-1} w_k exp\left(\frac{j2\pi nk}{K}\right) + d_n, 0 \le n \le K-1. \tag{3}$$

where *K* is the length of a frame, and w_k and d_n represent the frequency weighting and noise, respectively. Note that the noise is considered as an independent and identically distributed normal random variable with $N(0;\sigma^2)$.

The first term in the observation model, i.e., signal s_n, can be decomposed as

$$s_n = \frac{1}{K}\sum_{k=0}^{K-1} w_k exp\left(\frac{j2\pi nk}{K}\right) = \frac{1}{K}\left(w_0 + w_{K/2} + \sum_{k=1}^{K/2-1} \times \left\{w_k exp\left(\frac{j2\pi nk}{K}\right) + w_{K-k} exp\left(-\frac{j2\pi nk}{K}\right)\right\}\right). \tag{4}$$

As shown in (4), half of the complex bases are conjugate bases of the other half of the bases. If $w_k = w_{K-k}^*$ for $1 \le k \le K/2$-1, then a real signal is represented as

$$s_n = \frac{1}{K}\left(w_0 + w_{K/2} + 2\sum_{k=1}^{K/2-1} re\left(w_k\right) cos\left(\frac{2\pi nk}{K}\right)\right). \tag{5}$$

where *re*(·) is the operator for returning the real part of a complex number. Note that both w_0 and $w_{K/2}$ are real numbers because they represent DC and -DC components, respectively. From a neural network perspective, this can be implemented by considering the cosine function as an activation function in the last hidden layer, $re(w_k)$ as the weight connecting the last hidden layer and output layer, and $w_0 + w_{K/2}$ as a bias term in the output layer.

B. Network Architectures of Generator and Discriminator

This signal model is used to constrain the generator of a GAN network. We employ a MLP configuration where frequency bases are modeled in the first three layers (Fig. 1). In the first two layers, Rectified Linear Units (ReLUs) are applied for non-negative frequencies [25], while a cosine is applied as an activation in the third layer. Batch Normalization (BN) is applied to all layers [26]. The weights connecting to the output layer are set to real while w_k^{g4} and w_{K-k}^{g4} for $1 \le k \le K/2$-1 are set equal to each other to satisfy the weight constraint, $w_k^{g4} = w_{K-k}^{g4}$. Since a hyperbolic tangent is bounded within the interval $[-1, 1]$, it is applied to control the activation in the output layer. The sampling

TABLE I
THE LIST OF SCIENTIFIC NAMES FOR TARGET SPECIES AND ITS ABBREVIATION

Scientific Name	Abbreviation	The number of data segments (EA)
Hyla suweonensis	SuwFrog	5,303
Hyla japonica	GreFrog	1,664
Kaloula borealis	NarFrog	2,241
Rana dybowskii	BroFrog	676
Bombina orientalis	RedFrog	166

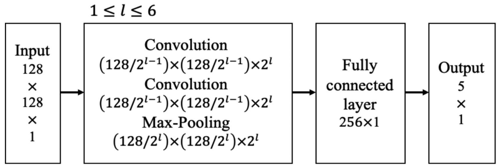
$1 \leq l \leq 6$
Input
128
×
128
×
1
Convolution
$(128/2^{l-1})\times(128/2^{l-1})\times 2^{l}$
Convolution
$(128/2^{l-1})\times(128/2^{l-1})\times 2^{l}$
Max-Pooling
$(128/2^{l})\times(128/2^{l})\times 2^{l}$
Fully
connected
layer
256×1
Output
5
×
1

Fig. 2. CNN network for calculating posterior probability in inception score and performing Amphibian sound classification test. The second block for convolution and pooling is stacked where *l* is a layer index.

TABLE II
INCEPTION SCORES DEPENDING ON OBJECTIVE FUNCTIONS

f_{GAN}	f_{LSGAN}	f_{WGAN} (c=0.005)	f_{WGAN_LP} (Λ_{LP}=15)
1.1483	2.3645	3.9795	2.3597

rate is set to 16 kHz, and the number of nodes in the intermediate layers is determined by considering the length of a frame to be 32 ms (512 samples).

The discriminator is designed based on a CNN whose input is formed by applying Short Time Fourier Transform (STFT) where the frame and sliding length are set to 32 ms and 8 ms, respectively. All convolutional filters are set to a size of 3 × 3 and the mid-layer feature maps are compressed by applying a max-pooling. After the fourth convolution and pooling, the 3D tensors are flattened to a vector, which is then connected to the output layer.

C. Adversarial Learning of Both Networks

An objective function for training the proposed architecture, f_{prop}. is defined as

$$f_{prop} = f_{WGAN} - \lambda \sum_{i,j} \left| w_{i,j}^{g4} \right|. \tag{6}$$

To understand the regularization on $w_{i,j}^{g4}$ further, it is important to note that the training data includes noises uncorrelated to the signals. Moreover, the network cannot distinguish between signal and noise during training. Thus, if the number of trainable parameters become sufficient to model both signal and noise, then the generator may also produce noisy waveforms. This can be considered as a type of overfitting in the generator. To resolve this issue, regularization is applied to the proposed objective function according to the assumption that amphibian calls are typically composed of a finite number of frequency bases. The training procedure uses the following settings: learning rate is set to 1.0e-6; batch size is 32; and the regularization coefficient is set to 1.0e-6.

IV. EXPERIMENTS

A. Database

In this letter, 5 types of amphibians indigenous to South Korea shown in Table I were chosen as target species. Their call sounds were manually collected in each natural habitat by five amphibian experts using a high-quality audio-recorder with 44.1 kHz sampling rate and stereo-recording. The call sound intervals for each species were annotated by the participants in the data collection. After converting to 16 kHz sampling rate and mono-type, the intervals were divided into 0.5 second segments by applying the endpoint detection method [27].

The database of real sounds was divided into a training set (*RealTrn*) and a test set (*RealTst*). The training set consisted of 150 randomly sampled audio clips for each class. The test set consisted of 200 randomly selected audio clips non-overlapping with the training set, with the exception of the RedFrog group. Given the rarity of this class, we used 150 clips for training and the full 166 audio clips for testing. While the great overlap in training and testing data for this specific class would result in an exaggerated performance, the purpose here is to compute an estimated baseline performance (Exp1) using the recorded dataset which is later used to compare performance with synthetic data.

B. Experimental Setting

1) Inception Score: Inception score is usually used for assessment of generative models by comparing their data distribution [16], [21], [28]. The posterior probability $p(y|x)$ is modeled by a CNN composed of convolution with a 3 × 3 filter, alternating 2D max-pooling, and fully-connected layer as in Fig. 2. As such, the synthesized stream is transformed to spectrogram by applying STFT with a frame of 16 ms with 75% overlap using a hamming window. Note that the CNN was also used in experiments for amphibian call sounds classification.

In this letter, the inception score is applied for assessments of four objective functions described in Section II for training the proposed architecture. Note that these functions were only applied without weight regularization. Also, a comparison of the proposed method to WaveNet and WaveGAN is performed by calculating the scores to demonstrate the effectiveness of the proposed method. To produce synthetic audios by WaveNet or WaveGAN, implementations publicly available on gitbub were used in this evaluation [20], [21]. Note that all audio generators were trained with the *RealTrn* set, by class.

2) Amphibian Sound Classification: Exp1: RealTrn sets are used for training and RealTst sets used for assessment as baseline. Exp2: Synthetic data produced by the proposed network are only used for training and the same *RealTst* sets used in evaluation. Exp3: Both the *RealTrn sets* and 600 synthetic data are used in training and the same *RealTst* sets used in evaluation. Note that all the classification rates are summarized by the average of results in 5 trials.

C. Experiment Results

1) Inception Score: In the first training step of the proposed networks, four objective functions for GAN [10], LSGAN [22], WGAN [23] and WGAN_LP [16] were used, and their inception scores are compared in Table II. As f_{WGAN} without gradient penalty loss gave the best performance, it is incorporated in (4) when designing the objective function of the proposed method.

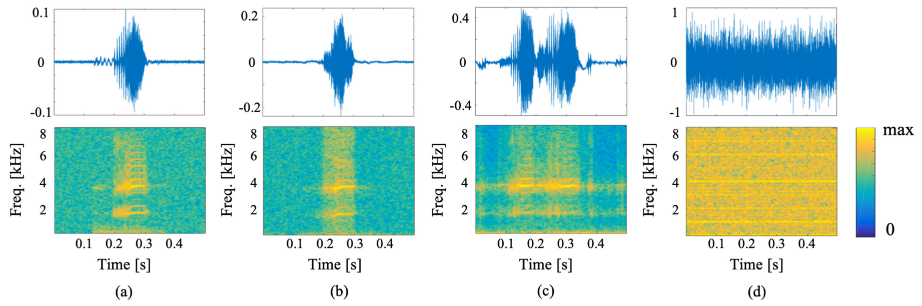

Fig. 3. Real and synthesized streams for SuwFrog according to methods (a) Real recording, (b) Proposed method, (c) WaveNet, (d) WaveGAN

TABLE III
INCEPTION SCORES OF DIFFERENT METHODS

WaveNet	WaveGAN	Proposed (Λ=1.0E-6 & *C*=0.005)
2.0257	1.1836	**5.4136**

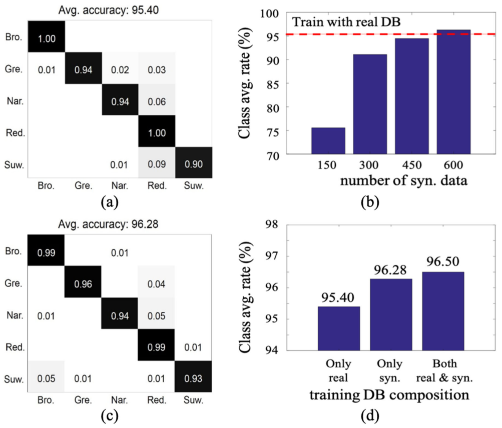

Fig. 4. Amphibian classification results. (a) confusion matrix of Exp1, (b) class avg. rate depending on the quantity of synthetic data used in training, (c) confusion matrix of Exp2 with 600 synthetic audios per a class, (d) class avg. rate depending on training data composition.

Table III shows assessments of audio stream generating methods in terms of the inception score. A combined objective function of a Wasserstein distance with weight clipping of c = 0.005 and a weight regularization step constitutes the overall proposed method, and its performance is compared to that of WaveNet and WaveGAN methods. WaveGAN is also designed based on WGAN, but it has a different architecture to the proposed network [21]. Fig. 3 shows an example of a real recording of SuwFrog and the versions synthesized by these methods. While WaveNet emulated real audio to a limited extent, WaveGAN seems to suffer from noise generation. The proposed method successfully manages to capture key aspects of audio signatures of real sound as shown in Fig. 3.

2) *Amphibian Sound Classification:* Exp1: The class average classification rate reached 95.40% and its confusion matrix is shown in Fig. 4(a). Exp2: The results depending on quantity of synthetic training data are summarized in Fig. 4(a). As expected, the more data is used during the training, the better performance is achieved. When 600 synthetic audio clips are used per class during training, the performance is comparable to that of Exp1. It is apparent that the distribution of the synthetic data closely matched that of the real data. The same overall result pattern is indicated by the confusion matrices for Exp1 and Exp2 is shown in Fig. 4(a) and (c). Exp3: This experiment investigates the effectiveness of data augmentation. The result represents an equal level for the result of Exp1, but the CNN converges faster when synthetic data is used in training as shown in Fig. 5. When the training data is augmented, processing time per epoch is longer than the case of using real only. However, the number of epochs required for convergence is significantly reduced by the addition of the synthetic data.

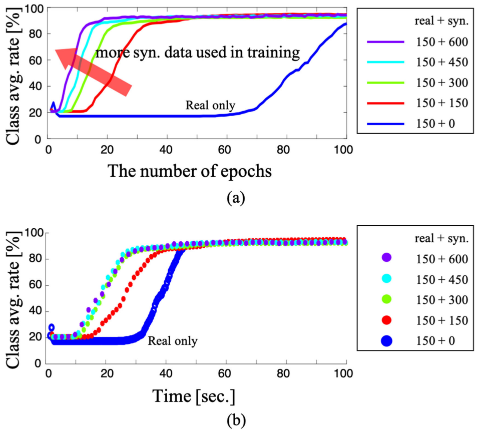

Fig. 5. Class avg. rate in training depending on training data composition. (a) class avg. rate over the number of epochs, (b) class avg. rate over the time.

V. CONCLUSION

This letter introduced a short-time audio generating network based on adversarial learning and investigated the effectiveness of data augmentation. The generator and discriminator were respectively designed based on MLP and CNN, and they were trained by maximizing Wasserstein distance with weight regularization. The effectiveness of the proposed method was experimentally demonstrated by measuring the inception score and performing classification test. The inception scores clearly demonstrated that the synthetic data closely resembles the target signal. Also, the results of amphibian classification using the CNN trained with synthetic data have shown that distribution of the synthetic data is very similar to the distribution of the real data. This investigation of data augmentation showed that the synthetic audios improved training efficiency when a combination of both the real and the synthetic data were used to train the classifier. Overall, these results demonstrated that the proposed network generates suitable amphibian sounds for data augmentation.

REFERENCES

[1] D. Hazell, J. M. Hero, D. Lindenmayer, and R. Cunningham, “A comparison of constructed and natural habitat for frog conservation in an Australian agricultural landscape,” *Biol. Conserv.*, vol. 119, no. 1, pp. 61–71, 2004.

[2] R. F. Baldwin, A. J. K. Calhoun, and P. G. deMaynadier, “Conservation planning for amphibian species with complex habitat requirements: A case study using movements and habitat selection of the wood frog rana sylvatica,” *J. Herpetol.*, vol. 40, no. 4, pp. 442–453, 2006.

[3] J. Colonna, T. Peet, C. A. Ferreira, A. M. Jorge, E. F. Gomes, and J. Gama, “Automatic classification of anuran sounds using convolutional neural networks,” in *Proc. ACM Int. Conf. Comput. Sci. Softw. Eng.*, 2016, pp. 73–78.

[4] J. Strout, B. Rogan, S. M. M. Seyednezhad, K. Smart, M. Bush, and E. Ribeiro, “Anuran call classification with deep learning,” in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2017, pp. 2662–2665.

[5] K. Ko, S. Park, and H. Ko, “Convolutional feature vectors and support vector machine for animal sound classification,” in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2018, pp. 376–379.

[6] M. A. Acevedo, C. J. Corrada-Bravo, H. Corrada-Bravo, L. J. Villanueva-Rivera, and T. M. Aide, “Automated classification of bird and amphibian calls using machine learning: A comparison of methods,” *Ecol. Inform.*, vol. 4, no. 4, pp. 206–214, 2009.

[7] P. Baldi, “Autoencoders, unsupervised learning, and deep architectures,” in *Proc. ICML Workshop Unsupervised Transfer Learn.*, 2012, pp. 37–49.

[8] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” 2013. [Online]. Available: http://arxiv.org/abs/1312.6114

[9] C. Doersch, “Tutorial on variational autoencoders,” 2016. [Online]. Available: http://arxiv.org/abs/1606.05908

[10] I. J. Goodfellow *et al.*, “Generative adversarial networks,” in *Proc. Advances Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[11] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, “Learning from simulated and unsupervised images through adversarial training,” in *Proc. Int. Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 2107–2116.

[12] J. Wang and L. Perez, “The effectiveness of data augmentation in image classification using deep learning,” 2017, *arXiv:1712.04621*.

[13] Q. Xu, Z. Qin, and T. Wan, “Generative cooperative net for image generation and data augmentation,” in *Integrated Uncertainty in Knowledge Modelling and Decision Making. IUKM 2019, (Lecture Notes in Computer Science)*, Cham, Switzerland: Springer, vol. 11471, 2019, pp. 284–294.

[14] G. Mariani, F. Scheidegger, R. Istrate, C. Bekas, and C. Malossi, “BAGAN: Data augmentation with balancing GAN,” 2018. [Online]. Available: http://arxiv.org/abs/1803.09655

[15] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, “Improved training of Wasserstein GANs,” in *Proc. Advances Neural Inf. Process. Syst.*, 2017, pp. 5769–5779.

[16] S. Augustin, “On the regularization of Wasserstein GANs,” in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–22.

[17] K. Roth, A. Lucchi, S. Nowozin, and T. Hofmann, “Stabilizing training of generative adversarial networks through regularization,” in *Proc. Advances Neural Inf. Process. Syst.*, 2017, pp. 2018–2028.

[18] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, “InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 2172–2180.

[19] S. Mun, S. Park, D. K. Han, and H. Ko, “Generative adversarial network based acoustic scene training set augmentation and selection using SVM hyper plane,” in *Proc. Workshop Detection Classification Acoust. Scenes Events*, 2017, pp. 93–97.

[20] A. van den Oord *et al.*, “WaveNet: A generative model for raw audio,” 2016. [Online]. Available: http://arxiv.org/abs/1609.03499

[21] C. Donahue, J. McAuley, and M. Puckette, “Adversarial audio synthesis,” in *Proc. Int. Conf. Mach. Learn.*, 2018. [Online]. Available: https://arxiv.org/abs/1802.04208

[22] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, “Least squares generative adversarial networks,” 2016. [Online]. Available: http://arxiv.org/abs/1611.04076

[23] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein GAN,” 2017. [Online]. Available: http://arxiv.org/abs/1701.07875

[24] J. Lin, “Divergence measures based on the Shannon entropy,” *IEEE Trans. Inf. Theory*, vol. 37, no. 1, pp. 145–151, Jan. 1991.

[25] V. Nair and G. E. Hinton, “Rectified linear units improve restricted Boltzmann machines,” in *Proc. Int. Conf. Mach. Learn.*, 2010, pp. 807–814.

[26] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” 2015. [Online]. Available: http://arxiv.org/abs/1502.03167

[27] J. Park, W. Kim, D. K. Han, and H. Ko, “Voice activity detection in noisy environments based on double-combined fourier transform and line fitting,” *Sci. World J.*, vol. 2014, 2014, Art. no. 146040.

[28] D. Berthelot, T. Schumm, and L. Metz, “BEGAN: Boundary equilibrium generative adversarial networks,” 2017, *arXiv:1703.10717*.