Synthesizing Game Audio Using Deep Neural Networks

Aoife McDonagh * †, Joseph Lemley * †, Ryan Cassidy†, and Peter Corcoran*
* College of Engineering and Informatics, NUI Galway, Galway, Ireland
† Xperi Corporation, Poway, CA, United States

***Abstract*—High quality audio plays an important role in gaming, contributing to player immersion during gameplay. Creating audio content which matches overall theme and aesthetic is essential, such that players can become fully engrossed in a game environment. Sound effects must also fit well with visual elements of a game so as not to break player immersion. Producing suitable, unique sound effects requires the use of a wide range of audio processing techniques. In this paper, we examine a method of generating in-game audio using Generative Adversarial Networks, and compare this to traditional methods of synthesizing and augmenting audio.**

I. Introduction

Immersion is an important aspect of gaming for creating a realistic and satisfying player experience. In the gaming community, immersion refers to the feeling of being cut off from reality, as if spatially located in the game environment with which one is interacting [1]. To provide an immersive experience, a game must capture the full attention of the player, allowing him/her to lose sense of time and self. The extent to which a player feels immersed appears correlated with the degree to which their visual, auditory and mental facilities are drawn into the game environment [2]. Auditory techniques for creating immersion have largely been underdeveloped compared to the level of investment in visual aspects of gaming. The quality of computer graphics is the largest factor influencing consumers when buying video games, according to a report by the Entertainment Software Association [3]. However, negative effects on player experience have been demonstrated when audio is removed entirely from a game [4]. System usability decreases because players receive fewer or no responses from the system in response to their actions and commands. Sense of presence is also degraded when a game is reduced to "nothing but animated graphics on a screen" [4]. While the importance of having audio in computer games is widely accepted, it is perhaps the effect of audio quality and depth that is less widely known and discussed. In this paper, we examine a method for the creation of unique sounds based on deep learning methods. We employ this technique to generate samples of in-game audio which could be used to improve in-game immersive experiences.

This research is funded under the SFI Strategic Partnership Program by Science Foundation Ireland (SFI) and FotoNation Ltd. Project ID: 13/SPP/I2868 on Next Generation Imaging for Smartphone and Embedded Platforms. This work is also supported Irish Research Council Employment Based Programme Awards EBP/2017/476 and EBPPG/2016/280

A. Deep Learning

Deep Learning has experienced significant interest in recent years, resulting in top performance on a number of important tasks. The concept of deep learning refers to a loose assortment of techniques, utilizing artificial neural networks (ANNs), that make use of two or more layers. The most popular of these methods include Convolutional Neural networks (CNN) [5] and Recurrent Neural Networks (RNNs) such as Long Short Term Memory (LSTM) [6] models.

The strength of ANNS, like other machine learning techniques, is that, unlike conventional computer programs, they can learn through example rather than by direct instruction. This allows them to learn things that humans understand intuitively but have a difficult time putting into concrete sets of procedures and rules [7]. Convolutional Neural Networks are heavily used in computer vision tasks, where they have recently surpassed human level accuracy on some benchmarks. A CNN is a type of artificial neural network that can learn features of datasets they are trained on in a way that provides translation invariance. This involves moving layers of small convolutional kernels over a matrix (such as an image) at a specific stride [8].

B. Generative Advesarial Networks

Generative Adversarial Networks [9] (GANs) are unsupervised machine learning models that learn the distribution of a dataset. They utilize two neural networks which are often, but not always, both implemented as CNNs. One of the networks, called the generator, takes random noise as input and learns to generate data that will fool a second network, called a discriminator. At the same time, the discriminator learns the binary classification task that classifies its input as "real", or "forgery" categories. In the case of images, the goal is for the generator to create images that seem to belong in the same distribution as the original images in the dataset [4]. For example, if the dataset was trained on pictures of cats, the generator would try to generate convincing pictures of cats. Although this method was initially designed for images, it can be used for any type of data, including audio, where it has recently generated considerable interest for speech and audio synthesis [10].

This paper explores the idea of using such a generative model, trained on raw audio data, to automatically generate

convincing in-game sounds that are unique to the game experience.

II. Related Work

A. Traditional Methods of Audio Synthesis

Audio synthesis has a rich history preceding the deep learning era. Common methods of augmenting audio signals range in complexity from simple pitch shifting to more sophisticated warping, equalization, and distortion techniques. Many of the traditional techniques have origins in analogue signal processing, where physical circuits were used to distort signals. Software has taken the place of much of the hardware previously used, having proved more flexible and cost effective. Audio engineers and composers are responsible for creating audio scenes and scores which meet the needs of a game. Considerable experience is required to utilize signal processing tools effectively, commonly amounting to years of specialized training. In large budget games, it is common for composers to record live audio for use in-game [11]. This is a large, laborious task requiring expensive equipment, well trained operators and sometimes the use of voice actors. In smaller budget indie games, composition of scores using electronic means is widespread [11]. Open source sound libraries provide another means of acquiring audio, somewhat inexpensively. Usually customization of pre-recorded audio is required to meet the acoustic needs of a game. This requires substantial post-processing work to be carried out by audio engineers, even though time is saved that would have otherwise been spent recording.

B. Deep Learning Methods of Audio Synthesis

Deep learning approaches to audio synthesis do not yet produce results as sophisticated as those seen in imaging tasks, but there have been some notable advancements in recent years. Inspired by generative models for images, WaveNet [12] is a deep neural network designed to generate raw audio waveforms. It has been applied successfully to speech generation tasks and music generation tasks as well as phoneme recognition when employed in a discriminative form. Baidu Research implements a variant of WaveNet in the audio synthesis block of their Deep Voice text-to-speech (TTS) system [13]. Deep Voice uses the same processing structure as traditional TTS systems but with all components replaced by neural networks trained on simple audio features like phonemes and fundamental frequency. These networks are easily adapted to new datasets, unlike traditional pipelines that require complicated tuning of hand-engineered features. Kalchbrenner et al. achieve faster than real-time audio synthesis using a sequential model, WaveRNN [13]. They also demonstrate that high fidelity audio generation is feasible on low-power mobile CPUs. This is particularly relevant to the gaming industry, where mobile gaming represents half of the global market [14].

III. Methodology

The goal of this research is to investigate methods of audio synthesis employing deep neural networks. Furthermore, we compare these methods with conventional methods of audio production. This comparison gives perspective on the applicability of deep learning to audio production in gaming.

A. Dataset

We use the Audio Set dataset [15] in the experiments outlined in this paper. It contains 1.7 million 10 second audio clips labelled on 632 audio event classes. The audio is taken from YouTube videos and has been labelled manually by humans. The dataset is available as CSV files identifying start and end times of each 10 second segment in a YouTube video or as 128-dimensional audio features stored as TensorFlow Record files.

B. Synthesis of audio using traditional methods

The most comparable method of audio synthesis to a generative adversarial network, in terms of output characteristics, is cross-synthesis. Taking two signals, a carrier and a modulator, a magnitude spectrum transform is performed to impress spectral characteristics of the modulator onto the carrier [16]. Some potential drawbacks of this approach when compared to the method proposed in this work are:

1) it is typically best when the carrier is broadband in quality, so that its energy can fully occupy the time-varying spectral envelope of the modulator, and
2) it is not clear how to easily extend this technique to more than two input signal types at a time.

Other methods of audio synthesis include augmenting an existing sound with transforms such as pitch shifting, time-stretching, or applying various filtering techniques. However, these are typically applied to a single sound of a single type (e.g., the sound of a dog barking), rather than combining multiple sounds of two or more signal types (e.g., combining dog sounds with speech sounds). We compare the performance of our GAN model against methods such as those mentioned here, in the context of synthesis of unique audio.

C. Synthesis of audio using GANs

Generative adversarial networks produce unique samples of audio by learning from a training data set. If this training data set contains multiple types of audio, it is possible that GAN outputs will exhibit characteristics of all these types morphed together in a hybrid fashion. This will achieve an effect similar to cross-synthesis, where information from two signals are combined into one. However, it is also readily noticed that this technique may be easily extended to generate hybrids of sounds from more than two types, by simply adding sounds of these types in equal or differing proportions in the training data set. Moreover, the relative proportion of a given type of sound in the set may be increased or decreased, in order to magnify or lessen the effect of that type of sound in the resulting hybrid output. In our experiments, we take a WaveGAN [10] model and train it on classes of audio from the Audio Set dataset.

The authors of the WaveGAN paper trained their network separately on multiple datasets with as little as 0.3 hours of data. A majority of Audio Sets classes contain at least as much audio, easily meeting our needs. We do not attempt to use the entire dataset, which contains over 5k hours of audio. For the experiments of this paper, without precluding our technique's ability to use sound types in unequal proportion, we aim, for simplicity of demonstration, to use as close to an equal proportion as possible between classes when multiple classes are included in the training data.

IV. Experiments

A. Duplication of WaveGAN experiments

The experimental steps outlined in the WaveGAN paper [10] are followed to verify the capabilities of the model. No hyperparameters are modified, and training is run for the same number of iterations (200000). For this experiment, training of a WaveGAN model is carried out on a subset of the Speech Commands Dataset [17].

B. Synthesis of a variety of data

An important question is whether WaveGAN can perform well synthesizing a variety of data. This includes data unlike what is seen in the original paper, as well as data which is an aggregation of multiple classes. The steps from the first experiment are applied to data from new sources and a mix of sources. Classes 'dog' and 'violin' are retrieved from AudioSet [15] and used individually and in combination to train WaveGAN models. The code used to retrieve and sort the AudioSet dataset is publicly available online .

Data from multiple classes, and without labels, is used to train a single model in this experiment. This examines the potential of a WaveGAN model to synthesize audio which sounds like a 'mixture' of multiple classes. Classes for this experiment were arbitrarily chosen by the authors.

C. Interpolation using latent space mapping

In the previous subsection, the problem of synthesizing unique audio by training on two or more classes to generate sounds that are hybrids of two or more classes was addressed. In that approach, random latent vectors are input to the generator to create a sound sample with the intent that some subset of these samples will have the desired audio properties (ie. a violin sound that has the rhythm of a dog bark).

Once a generator has been trained, it is also possible to learn the reverse mapping. This mapping allows one to find an input vector that, when given to a GAN, will produce a sound closest to a provided sound sample. By using this technique, it is possible to interpolate between two or more sounds in the latent space. This inverse latent space mapping has been used successfully in computer vision tasks [18].

In this experiment, interpolation in the latent space is used to generate a range of sound samples between two chosen sounds.

https://github.com/aoifemcdonagh/audioset_processing

V. Results and Evaluation

The quality of generated audio varies for each set of data used to train the model. Generally, output from a model trained on data from a single class is superior to output from a model trained on data from multiple classes. It is unsurprising that there would be some difficulty with reproducing samples from a data set where there are multiple classes present but no explicit labels associated with them. Exact reproduction is also complicated by the random input vectors, which means the same dataset could result in a different slightly different set of sounds each time a network is trained on them.

Another factor that contributes to the output of the model is the quality of the training data. The numerical digits subset of the Speech Commands Dataset [17] (SC09) is a set of high quality data with little background noise. Each example in this set contains only one digit. On the other hand, AudioSet [15] examples can contain sound from multiple sources (classes). Problems arise when, for example, a ten-second clip contains only 1 second of the desired audio class, or when a significant amount of mis-labeled data makes it into the set used for training. These factors could contribute to the noisy, somewhat chaotic sounding output from models trained on data from AudioSet.

Generative models are notoriously difficult to evaluate due to the subjective nature of their output. The 'realness' of a generated image or piece of audio can be judged instinctively by a human observer, yet remains an almost impossible task for a machine. There are no robust techniques to objectively measure generative model output quality apart from some ad-hoc methods like Inception Scoring [19]. This technique has fundamental flaws which are outlined in recent work [20] and by the creators of WaveGAN [10] itself. Due to the shortcomings in qualitative evaluation metrics, the best method of evaluating generated samples is currently to listen to them. A more rigorous approach that does not contradict human judgment has yet to be developed and remains an important aspect of future work.

A subjective scoring of each models output was performed by a human listener. A model's score at a given training iteration is given by the percentage of output samples which resemble the class of data it is trained on. The scores for each model are given below in Table I. For the model trained on two audio classes, a pass is given when the output sample resembles either of the original classes or is considered a credible mix of the two.

Model	100k	150k	200k	250k
SC09	63%	53%	75%	-
Dog	50%	41%	28%	-
Violin	28%	28%	47%	-
Dog & Violin mix	34%	34%	44%	25%

TABLE I: Pass rates of models at various stages of training (given in number of iterations)

An interesting contradiction occurs when considering what constitutes as a 'pass' for the model trained on the digits subset

of the Speech Commands Dataset[17] (SC09). Some examples produced by the model resemble mixtures of multiple digits. In this case, mixed words are undesirable outcomes, because a listener would expect to hear one of the ten digits in the training data. However, in the case of our 'mixed' model, the goal is to produce samples which exhibit characteristics of all classes included in the training data. This anomaly supports the idea that GANs can be used to synthesize audio with multi-class characteristics as a desirable trait. For the purposes of comparing with the samples provided by the creators of WaveGAN [10] in the case of the SC09 dataset, a pass is considered to be an output sample which resembles a single digit appearing in this set.

Our methods, which employed a single GAN model, performed well on multiple datasets. This is an encouraging step as it is common for GANs to perform poorly when a new dataset is used without hyperparameter tuning.

VI. Conclusions

Through training a generative adversarial network on a set of raw audio containing data from multiple classes, audio samples are produced that exhibit combined features of these classes. Furthermore, using the latent space of the deep learning audio synthesis model, audio samples were produced that exhibit features of multiple audio classes. The resulting sound files will be made available to attendees at IEEE GEM Conference 2018.

It has been shown that GANs can be used to generate a wide range of audio without relying on spectrograms or predefined filters. The experiments in this paper demonstrate that they can also be used for synthesis of novel audio, although subjective sound quality could be improved and objective assessment remains an open problem.

The methods applied in this paper are applicable to the production of audio for games, particularly in the case where sound effects need to sound unlike anything existing in reality. Such a need can arise in games with fantasy or extra-terrestrial themes where it could break the immersive sensation if characters and objects sound overly earth-like.

References

[1] C. Jennett, A. L. Cox, P. Cairns, S. Dhoparee, A. Epps, T. Tijs, and A. Walton, "Measuring and defining the experience of immersion in games," *International Journal of Human-Computer Studies*, vol. 66, no. 9, pp. 641–661, September 2008.

[2] E. Brown and P. Cairns, "A grounded investigation of game immersion," in *Extended abstracts of the 2004 conference on Human factors and computing systems - CHI '04*. New York, New York, USA: ACM Press, 2004, p. 1297.

[3] ESA, "2007 Essential Facts about the Computer and Video Game Industry," The Entertainment Software Association, Tech. Rep., 2017. [Online]. Available: http://www.theesa.com/article/2017-essential-facts-computer-video-game-industry/

[4] K. Jørgensen, "Left in the dark: playing computer games with the sound turned off," in *From Pac-Man to Pop Music: Interactive Audio in Games and New Media*. Ashgate, 2008, ch. 11, pp. 163–176.

[5] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[6] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[7] J. Lemley, S. Bazrafkan, and P. Corcoran, "Deep Learning for Consumer Devices and Services: Pushing the limits for machine learning, artificial intelligence, and computer vision." *IEEE Consumer Electronics Magazine*, vol. 6, no. 2, pp. 48–56, 2017.

[8] A. Goodfellow, Ian, Bengio, Yoshua, Courville, *Deep Learning*. MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org/

[9] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Networks," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[10] C. Donahue, J. McAuley, and M. Puckette, "Synthesizing audio with generative adversarial networks," *CoRR*, vol. abs/1802.04208, 2018. [Online]. Available: http://arxiv.org/abs/1802.04208

[11] B. Schmidt, "GameSoundCon Game Audio Industry Survey 2017," GameSoundCon, Tech. Rep., 2017. [Online]. Available: https://www.gamesoundcon.com/single-post/2017/10/02/GameSoundCon-Game-Audio-Industry-Survey-2017

[12] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *CoRR*, vol. abs/1609.03499, 2016. [Online]. Available: http://arxiv.org/abs/1609.03499

[13] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A. van den Oord, S. Dieleman, and K. Kavukcuoglu, "Efficient Neural Audio Synthesis," feb 2018. [Online]. Available: http://arxiv.org/abs/1802.08435

[14] Newzoo, "Global Games Market Report," Newzoo, Tech. Rep., 2018. [Online]. Available: https://newzoo.com/insights/articles/global-games-market-reaches-137-9-billion-in-2018-mobile-games-take-half/

[15] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio Set: An ontology and human-labeled dataset for audio events," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, mar 2017, pp. 776–780.

[16] C. Roads and J. Strawn, "Linear Predictive Coding," in *The Computer Music Tutorial*. MIT Press, 1996, ch. 2, pp. 200–212.

[17] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," *CoRR*, vol. abs/1804.03209, 2018. [Online]. Available: http://arxiv.org/abs/1804.03209

[18] S. Bazrafkan, H. Javidnia, and P. Corcoran, "Face synthesis with landmark points from generative adversarial networks and inverse latent space mapping," *arXiv preprint arXiv:1802.00390*, 2018.

[19] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems*, 2016, pp. 2234–2242.

[20] S. Barratt and R. Sharma, "A note on the inception score," *arXiv preprint arXiv:1801.01973*, 2018.