**Generative Adversarial Networks: Voice Synthesis**

*Harshvardhan Singh*

*Department of Computer Science and Engineering,*

*SRM Institute of Science and Technology, Kattankulathur, India, 603203*

*e-mail: hs7805@srmist.edu.in*

*Shubham Samanta*

*Department of Computer Science and Engineering,*

*SRM Institute of Science and Technology,*

*Kattankulathur, India, 603203*

*e-mail:* ss5098@srmist.edu.in

*S. S. Saranya*

*Department of Computer Science and Engineering,*

*SRM Institute of Science and Technology, Kattankulathur, India, 603203*

*e-mail: saranyas6@srmist.edu.in*

**Abstract**

The Generative Adversarial Network framework is an extremely powerful and promising deep neural network framework that has yet to reach its full potential. GAN Networks come in a variety of configurations and are capable of creating a variety of data types, including images, audio, and three-dimensional objects. However, in our study, we are interested in the generation of human voices via GAN-based Speech-To-Speech synthesis. Previous research has successfully generated not only human voices, but also animal and instrument sounds. Despite this, the generated voices frequently sound artificial and unpolished due to the existence of a humming sound. In this project, we construct and train a system capable of synthesizing a person's voice in real time using our developed generative adversarial network (GAN). Our algorithm is trained on an audio dataset. Our dataset consists of a corpus of brief audio clips of various individuals reciting the same lines. We have improved the current voice cloning technology by enhancing the generated voice's quality by eliminating the humming sound that is present in the created speech and by making it sound less robotic. We have been successful in closely matching the pitch and frequency of the speech in the audio data to the target voice.

**Keywords**

Generative Adversarial Network, Feature Extraction, Spectrogram, Generator, Discriminator, Post Training

**Introduction**

Voice cloning has the potential to be a game-changing project that would benefit people all over the world in a variety of fields. One of the applications of the research would be the reproduction of the voices of celebrities and performers who have passed away and are no longer with us, such as Elvis Presley and Frank Sinatra. While it is possible that this technology will have some bad applications, we must not let this deter us from investigating this exciting technology that has such much promise for progress. Voice cloning is accomplished through the use of a sort of Generative Adversarial Network (GAN) that has been referred to as waveGAN in prior studies and is very similar to our model; this type of network is used in our approach. While waveGAN operates on a 2D Time-Frequency waveform, our discriminator makes use of spectrograms to make its determination. Our GAN model requires two audio samples to be trained before it can be used. It is decided which of the audio samples will be used as the target voice, and the other sample is adjusted so that it matches and sounds exactly like the target sample. In order to accomplish this, functions from Python libraries such as "Wave," "Librosa," or "Pydub" are used to alter the pitch and frequency of the underlying audio sample.

**Dataset Description**

Our audio dataset is used as a training set to train a GAN model, which is then used to generate output. A corpus of voices from different persons all saying the same phrases makes up the piece. The audio recordings are of people of both sexes, from a variety of nations, with a variety of accents, who are speaking.

**Architecture of our GAN model**

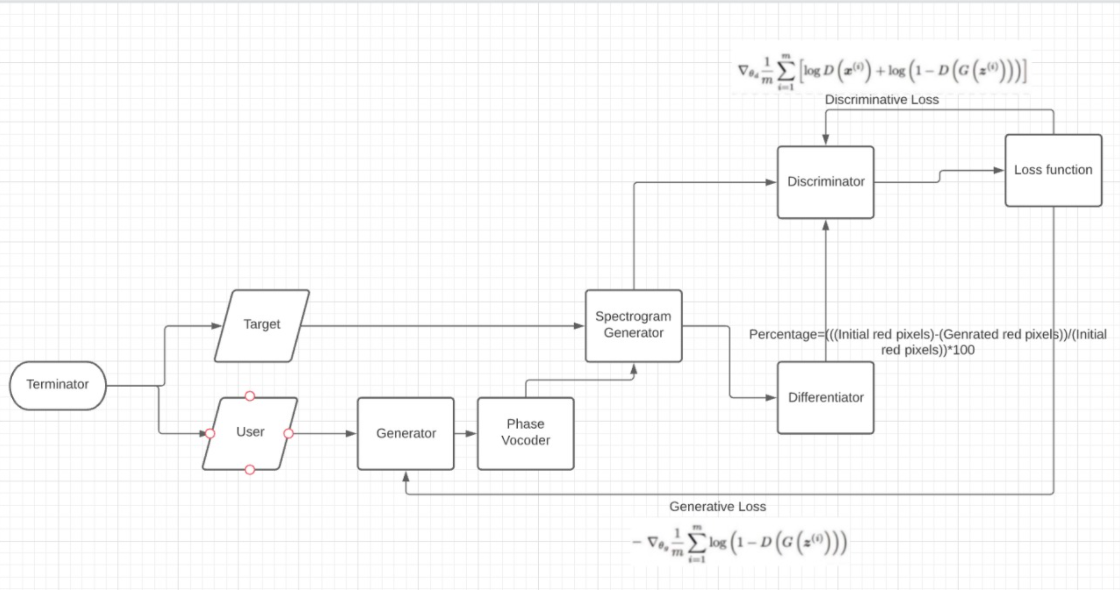
****

Fig 1. Architecture diagram

The process starts with accepting two wav files, one containing the target voice saying a word and the other wav file containing user voice saying the same word.

During the first loop the generator will not make any changes. The Spectrogram then makes the spectrogram of the two wav files and send It to the differentiator and the discriminator.

After the discriminator has produced results using the Sigmoid function and the Differentiator has given its results (0% in the first iteration), the data is then sent to the loss function generator.

The loss function then generates a generative and discriminative loss and feeds it to generator and discriminator respectively.

From the second iteration the generator makes changes in the wav file of the user according to the generative loss it has received from the previous iteration.

The generator with the help of the phase Vocoder Changes the pitch of the wav file given to it by the generator without changing the length of the wav file.

The generated wav file from the phase vocoder is then sent to the spectrogram generator.

The spectrogram generator then generates the spectrogram of the altered wav file it received from the phase vocoder and then sends it to the discriminator and the differentiator.

The differentiator the uses the formula in the above diagram to determine the progress made by the generator. It tags the highly varying pixels in red. The no. of red pixels present generated by the differentiator can then be used to calculate the progress made by the generator.

The discriminator then tries to discriminate between the target spectrogram and the altered voice spectrogram.

The result from both differentiator and discriminator is then fed to the loss function which generates the generative and discriminative loss function for the next iteration and returns control to the start.

The process would be repeated until and unless there is 80% improvement given by the differentiator.

**Feature Extraction and Spectrogram Generation**

A spectrogram is a pictorial representation of the frequency spectrum of sound. It takes into account the amplitude (volume), the frequency, and the pitch. We chose to build spectrograms of our audio dataset rather than wavelet representations since it allows us to better distinguish the timbre of a sound as well. This element of sound has assisted us in distinguishing between the voices of two individuals, which can be identified using an image discriminating code rather than the time-amplitude graph of sound. The spectrogram proved to be more useful because it contains more elements of sound, which let us identify which portions of the input voice needed to be tweaked. It was also necessary to apply the Short-Time Fourier Transform (STFT) in order to generate a spectrogram from an audio file, because the simple vector of the voice had to be divided into two vectors: a time vector and a frequency vector.

**Generative Adversarial Network**

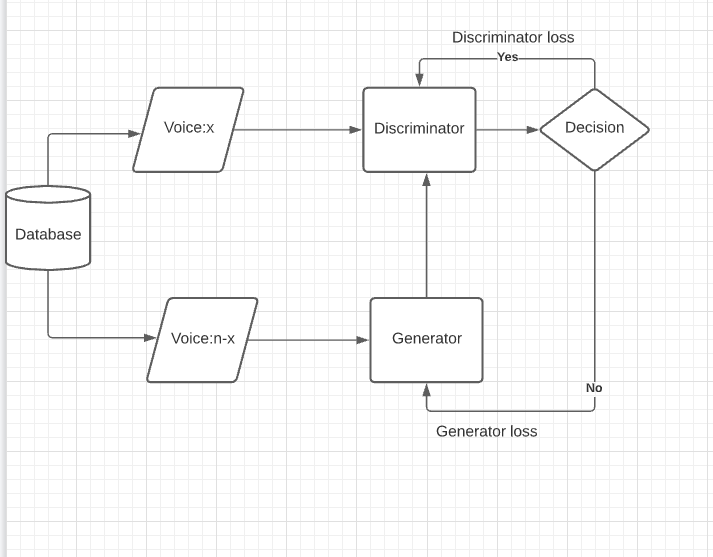
****

Fig 1. Architecture of GAN training

Generative Adversarial Network is basically a set of two neural networks fighting each other which are the generator and the discriminator. This generates values in order to tweak the spectrogram and the discriminator tries to discriminate which input is a fake spectrogram. The Generator in this case tweaks the frequency, pitch, loudness of the input voice in order to mimic the spectrogram of the target voice. The generator tries to discriminate which spectrogram is the true spectrogram of the target voice. With time, both neural networks improve as the data set grows. This gives Generator an adversary to overcome, which too learns with time and experience giving us great results. In our model, we used a pre-trained discriminator from super-enhancer GAN thus cutting the time to train the Generator. Moreover, in our case, the generator works with audio files and the discriminator workings with spectrograms/ image files.

**Post-Training GAN**

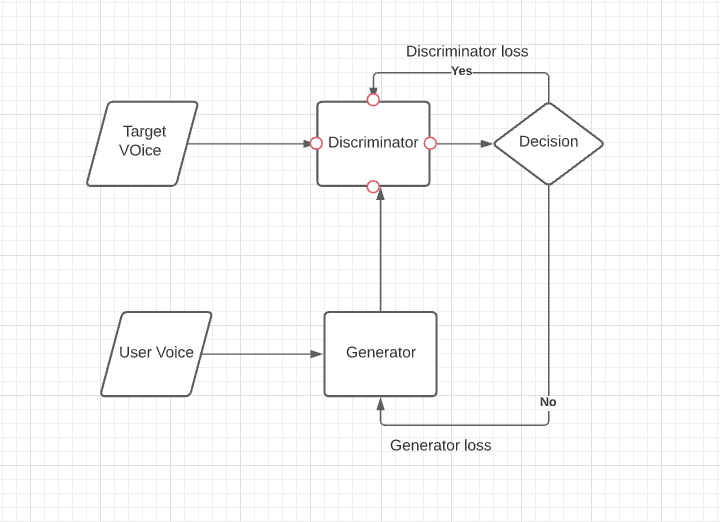
****

Fig. Architecture of post training

After the training phase is done, we first input an audio file of a couple of seconds. This would be our target voice. Subsequently, the user would enter an audio file of his voice pronouncing the same word. The generator and the Discriminator would start training and fighting each other. In the end, the generator would output a set of values for loudness, pitch, frequency, and timbre difference between the two audio files which we can directly use to manipulate the audio of the user according to the generator at the end process.

**Observation**

We have successfully developed a model that is capable of speech-to-speech synthesis. The voices generated matches the target voice and there is minimum noise. The discriminator, which is a CNN based binary classifier gives an output of 0.87 which indicates that the generated output closely matches the original voice with an approximate 90% accuracy.

We noted the average discriminative values for 5 voices, and have tabulated them below:

|  |  |  |
| --- | --- | --- |
| S. No. | Epochs | Discriminative value |
| 1. | 100 | 0.13554838 |
| 2. | 200 | 0.35587492 |
| 3. | 350 | 0.44594881 |
| 4. | 400 | 0.51203934 |
| 5. | 500 | 0.59583745 |
| 6. | 600 | 0.63568374 |
| 7. | 700 | 0.69447212 |
| 8. | 800 | 0.74654356 |
| 9. | 900 | 0.77694835 |
| 10. | 1000 | 0.81698457 |
| 11. | 1100 | 0.82345845 |
| 12. | 1200 | 0.84564356 |
| 13. | 1300 | 0.86565434 |
| 14. | 1400 | 0.86849351 |

Table 1: Discriminative Values

Graph 1: Discriminative values for different number of iterations during GAN training

**Result**

We were successful in generating voice that closely resembles the voice of an actual person after 1400 iterations. The voice generated is clear and does not sound too robotic.

**References**

[1] “Live DeepFake”, Yisroel Mirksy, Oleg Brodt, Joshua Cohen, Ruven Levy, Itay Blokh, journal of creating gans and artificial intelligence, august 2019

[2] “Conditional WaveGAN”, Chae Young Lee, Anoop Toffy, Gue Jun Jung SK, Woo-Jin Han Netmarble, arXiv:1809.10636v1 [cs.CV] 27 Sep 2018

[3] “SINGAN: Singing voice conversion with Generative Adversarial Networks”, Berrak Sisman, Karthika Vijayan, Minghui Dong, Haizhou Li, IEEE, 2019

[4] “Synthesising Audio with Generative Adversarial Networks”, Marco Pasini, 2019,

[5] “Voice Translation and Audio Style Transfer with GANs”, Marco Pasini, 2019,

[8] “Domain speech recognition using nonparallel corpora with cycle-consistent adversarial networks”, Masato Mimura, Shinsuke Sakai, and Tatsuya KawaharaCross IEEE, 2017.

[9] “Applying Voice Conversion to Concatenative Singing-Voice Synthesis”, Fernando Villavicencio and Jordi Bonada, INTERSPEECH, 2010

[10] W. Ping, K. Peng, and J. Chen, “Clarinet: Parallel wave generation in end-to-end text-to-speech,” 2018.

[11] R. Prenger, R. Valle, and B. Catanzaro, “Waveglow: A flow-based generative network for speech synthesis,” 2018.

[12] D. P. Kingma and P. Dhariwal, “Glow: Generative flow with invertible 1x1 convolutions,” 2018.

[13] R. Yamamoto, E. Song, and J.-M. Kim, “Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” 2019.

[14] K. Kumar, R. Kumar, T. de Boissiere, L. Gestin, W. Z. Teoh, J. Sotelo, A. de Brebisson, Y. Bengio, and A. Courville, “Melgan: Generative adversarial networks for conditional waveform synthesis,” 2019. [15] S. O. Arik, M. Chrzanowski, A. Coates, G. Diamos, A. Gibiansky, ¨ Y. Kang, X. Li, J. Miller, J. Raiman, S. Sengupta, and M. Shoeybi, “Deep voice: Real-time neural text-to-speech,” CoRR, vol. abs/1702.07825, 2017.

[16] Y. Wang, R. J. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. V. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, “Tacotron: A fully end-to-end text-tospeech synthesis model,” CoRR, vol. abs/1703.10135, 2017.

[17] S. O. Arik, G. F. Diamos, A. Gibiansky, J. Miller, K. Peng, W. Ping, ¨ J. Raiman, and Y. Zhou, “Deep voice 2: Multi-speaker neural text-tospeech,” CoRR, vol. abs/1705.08947, 2017.

[18] W. Ping, K. Peng, A. Gibiansky, S. O. Arik, A. Kannan, S. Narang, ¨ J. Raiman, and J. Miller, “Deep voice 3: 2000-speaker neural text-tospeech,” CoRR, vol. abs/1710.07654, 2017.

[19] H. Shintel, N. Anderson, and K. Fenn, “Talk this way: The effect of prosodically conveyed semantic information on memory for novel words,” Journal of experimental psychology. General, vol. 143, 04 2014.

[20] K.-L. Huang, S.-F. Duan, and X. Lyu, “Affective voice interaction and artificial intelligence: A research study on the acoustic features of gender and the emotional states of the pad model,” Frontiers in Psychology, vol. 12, p. 1212, 2021.