SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379 / https://doi.org/10.14445/23488379/IJEEE-V10I2P101

Volume 10 Issue 2, 1-7, February 2023

Original Article

Automatic Drum Beat Generation using GAN

Suman Maria Tony[1], S. Sasikumar[2]

[1, 2]*Department of Electronics and Communication Engineering, Hindustan Institute of Technology and Science, Chennai, India*

[1]*Corresponding Author: sumanmariatony@licet.ac.in*

Received: 21 December 2022 Revised: 01 February 2023 Accepted: 10 February 2023 Published: 28 February 2023

Abstract *- In this paper, the authors attempt to automatic generation of drum beats using generative adversarial networks (GAN). The generator of the GAN is trained with the short-time Fourier to transform (STFT) of drum beats from a diversified dataset, while the discriminator challenges the generator. The generator, once trained, the GAN is able to produce drum beats close to real-time sequences. Also, we propose to do a subjective evaluation of the generated drum beats. The simulation results showed that the drum beat generated by the GAN had more resemblance when compared to the actual drum beats. Also, the subjective assessment by a few audiences proves the effectiveness of this method of automatic drum beat synthesis.*

Keywords *- Music generation, Generative adversarial networks, Drum beats, Music synthesis introduction.*

1. Introduction

Since the development of potential deep neural networks, researchers have been attempting to generate music artificially. The generation is done in supervised or unsupervised modalities. There is significant work using machine learning in imaging as well as in other perspectives. Automating this music generation for a given scenario is a complex task where no scientific community has achieved results on par with human composition. The idea is to generate a melody or a polyphony without any chords or patterns. The digitized song is converted to the musical instrument digital interface (MIDI) format to ease the processing and synthesizing of music data.

1.1. Literature Review

A significant work in audio generation using genetic algorithms has been carried out for many years [1, 2]. Here, time series networks like Recurrent Neural Networks (RNN) play a major role in predicting and synthesizing music sequences [3]. GAN has become popular due to its ability to produce quality images when trained for a particular task [4]. GAN, in its other forms, can also generate images close to the real images of humans [5]. GAN, nowadays, is used for generating symbolic music [13, 14] and audio files [15, 16]. There is significant work for music generation using GAN and RNN with reinforcement learning [6, 7] and convolutional networks [8, 9]. Several such automated music composition algorithms based on RNN were developed for easy composition focusing on the time to deployment. A few such systems available online are Magenta [17], DeepJazz [18], BachBot [19], FlowMachines [20], and WaveNet [21].

In this paper, we attempt to generate drum beats using pre-trained GAN models after a few modifications. The GAN model is trained with the STFT of several drum beats instead of the direct audio, and the GAN is expected to generate the same. Later, the generated drum STFT can be converted into actual drum beats and verified for quality. We compare the outcomes of the DCGAN–based [23, 25, 28] algorithms, namely SpecGAN and WaveGAN, proposed by Donahue et al. [24]. The preliminary work carried out by Suman et al. [26, 34] concentrated on LSTM and GAN for music synthesis, and significant work was reported in speech synthesis [27].

The mathematical formulation of GAN is given as

$$\min_{Ge}\max_{Di} V(Ge, Di) = log(De(x)) + log(1 - De(Gi(z))) \quad (1)$$

$V(Ge, Di)$ is the target function in which De should be maximized and Gi should be minimized.

$Di(x)$ is the probability that the input x is taken from the original data as classified by Di

$Ge(z)$ is the output of Ge with z random input noise data

The discrete STFT of a signal is expressed by equation (2). Here the discrete signal is broken into finite frames, and a Fourier transform is applied to it.

$$STFT\{x[n]\}(m, \omega) \equiv X(m, \omega) = \sum_{n=-\infty}^{\infty} x[n]w[n - m]e^{-j\omega n} \quad (2)$$

X[n] is the signal, and w[n] is the window.

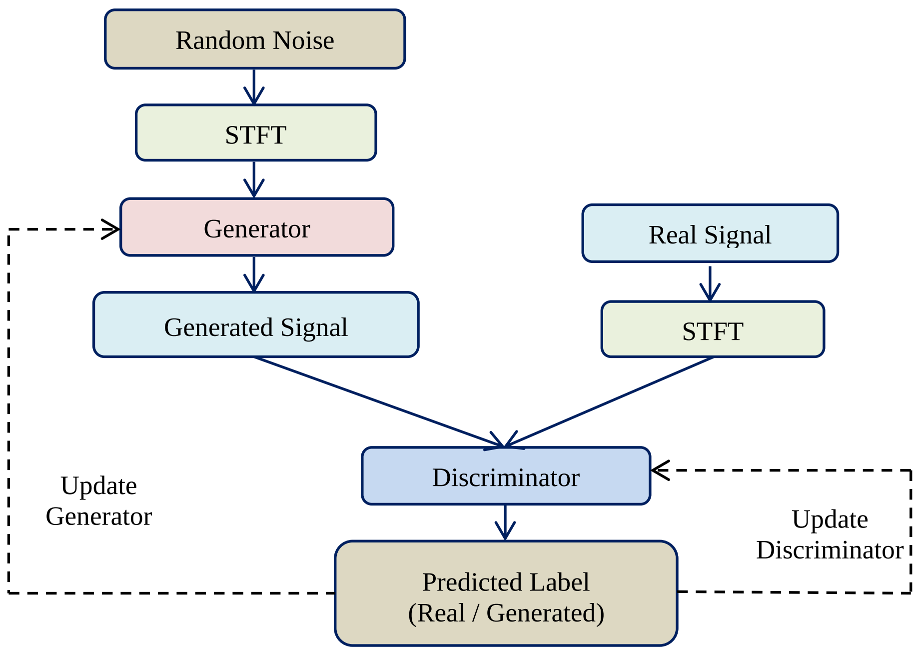


Fig. 1 GAN topology for drum beat synthesis

The spectrogram is computed by squaring the STFTs, which are used to plot the spectrogram of the synthesized drum beat later in this paper in chapter 5.

$$spectogram\{x[t]\}(\tau, \omega) \equiv |X(\tau, \omega)|^2 \quad (3)$$

Section 1 introduces the challenges in music synthesis, especially the work carried out in drum beat generation and GAN. Section 2 highlights the basic blocks of GAN for synthesizing drum beats with the help of STFT. Section 3 explains the GAN architecture, various parameters and methods used to conduct the experiment. The training of GAN and the synthesis of drum beats is discussed in section 4. The simulation results, experiment outcomes, and subjective evaluation are discussed in section 5. The concluding remarks are given in section 6.

2. GAN for Drum Beat Synthesis

In this method, the training data, STFT of the real drum beat, is computed, and a huge library is created. The discriminator further uses this training data to find the difference between the generated signal and the STFT of the original drum beat. The generator tries to generate data similar to the STFT of the original drum beat from a random vector or data. As shown in Figure 1, the discriminator classifies its input as real or fake from the input provided by the generator and the original drum beats.

The generator's objective is to fool the discriminator or synthesize drum beats very close to the original one so that the discriminator identifies or classifies it as real. Ultimately, the fake STFTs of drum beats close to the real ones are generated. The loss of the discriminator is maximized to improve the performance of the generator, while to improve the performance of the discriminator, its loss needs to be minimized. This trade-off determines the training and synthesis of drum beats.

3. Materials and Methods

3.1. GAN Architecture

The generator network shown in Figure 2 generates STFTs from a 1X100 random array. This network's outcome is an array that is supposed to be close to the STFT of the real data. The random data generator is followed by a fully connected layer for up-scaling the random data to a 128X128X1 array. Following that, we have transposed convolution layers, ReLU layers and a tanh as the last layer. The generator network structure and the layer dimensions are given in Table 1.

3.2. GAN Parameters

The discriminator used here is shown in Figure 2, and it takes a 128X128 image and predicts the score using a series of layers with leaky-ReLU (LReLU) and a fully connected layer. The discriminator network structure and the layer dimensions are given in Table 2.

4. Training and Synthesis

In this experiment, the discriminator uses the drum beat's real audio signal. The discriminator is trained to identify real STFTs and fake STFTs, as shown in Figure 3. Perhaps, the generator is trained to create fake STFT representations of drum beats. The real drum beats were used to compute the STFT, and those files form the base dataset.

Table 1. Generator network structure for drum beat synthesis

Operation	Kernel	Output Shape
Input z ≈ Uniform (-1;1)		(n, 100)
Dense 1	(100, 256d)	(n, 256d)
Reshape (n, 4, 4,16d)		
ReLU		(n, 4, 4, 16d)
Trans Conv2D (Stride=2)	(5, 5, 16d, 8d)	(n, 8, 8, 8d)
ReLU		(n, 8, 8, 8d)
Trans Conv2D (Stride=2)	(5, 5, 8d, 4d)	(n, 16, 16, 4d)
ReLU		(n, 16, 16, 4d)
Trans Conv2D (Stride=2)	(5, 5, 4d, 2d)	(n, 32, 32, 2d)
ReLU		(n, 32, 32, 2d)
Trans Conv2D (Stride=2)	(5, 5, 2d, d)	(n, 64, 64, d)
ReLU		(n, 64, 64, d)
Trans Conv2D (Stride=2)	(5, 5, d, c)	(n, 128, 128, c)
Tanh		(n, 128, 128, c)

Table 2. Discriminator network structure for drum beat synthesis

Operation	Kernel	Output Shape
Input x or G(z)		(n, 128, 128, c)
Conv2D (Stride=2)	(5, 5, c, d)	(n, 64, 64, d)
LReLU (α=0.2)		(n, 64, 64, d)
Conv2D (Stride=2)	(5, 5, d, 2d)	(n, 32, 32, 2d)
LReLU (α=0.2)		(n, 32, 32, 2d)
Conv2D (Stride=2)	(5, 5, 2d, 4d)	(n, 16, 16, 4d)
LReLU (α=0.2)		(n, 16, 16, 4d)
Conv2D (Stride=2)	(5, 5, 4d, 8d)	(n, 8, 8, 8d)
LReLU (α=0.2)		(n, 8, 8, 8d)
Conv2D (Stride=2)	(5, 5, 8d, 16d)	(n, 4, 4, 16d)
LReLU (α=0.2)		(n, 4, 4, 16d)
Reshape		(n, 256d)
Dense	(256d, 1)	(n, 1)

Now, sufficient training makes the generator generate drum beat sequences similar to the real one. As the generated sequence is in the time-frequency domain, an inverse STFT yields the generated audio drum beat signal, as shown in Figure 4. The training process involves 1000 epochs in general and is computationally intensive.

5. Result and Discussion

Training GAN with a sufficient dataset of original drum beat files in batches yields good results in synthesizing the same kind. Figure 5 shows the generated drum beats in the time domain, and the STFT of the same. Originally, STFTs were generated, and ISTFT gives the audio file, which can be played as a drum beat. A sample of 5 drums beat STFTs and their equivalent time domain representation are given here.

From the figure, it is clear that the discontinuity is largely avoided, and by listening to the generated music file, it can be scored by as many listeners to assess the quality of the synthesized drum beat qualitatively.

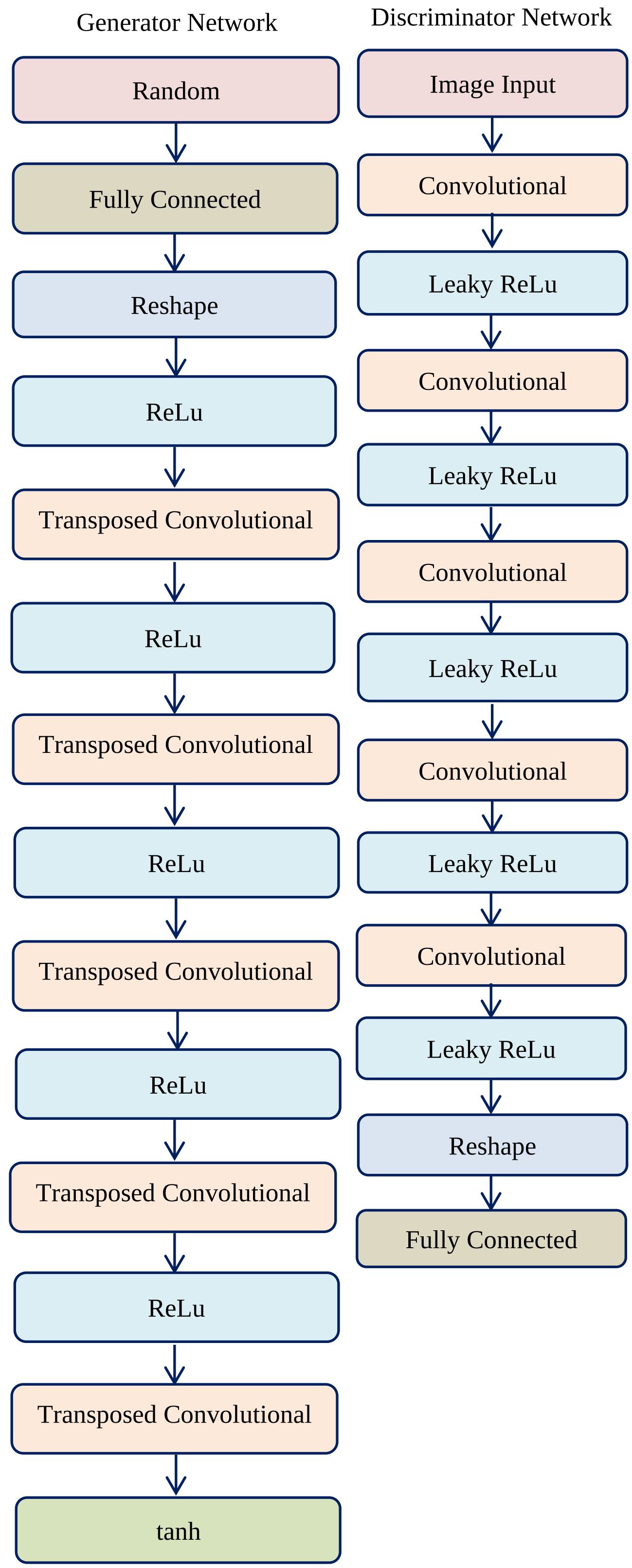


Fig. 2 GAN architecture: generator and discriminator

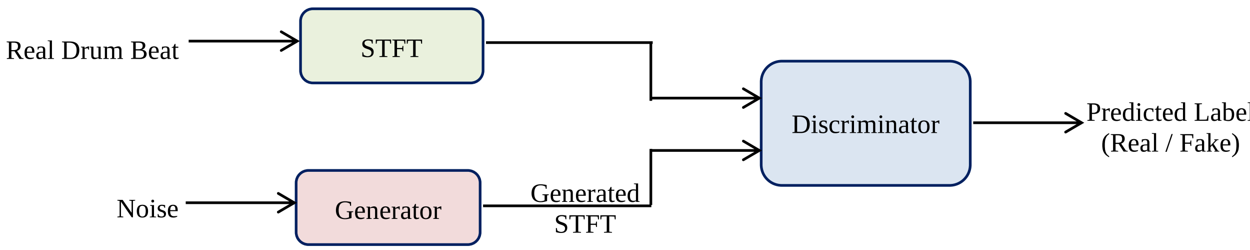


Fig. 3 GAN training

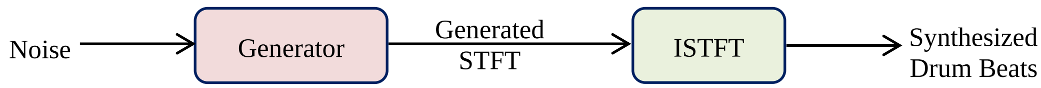
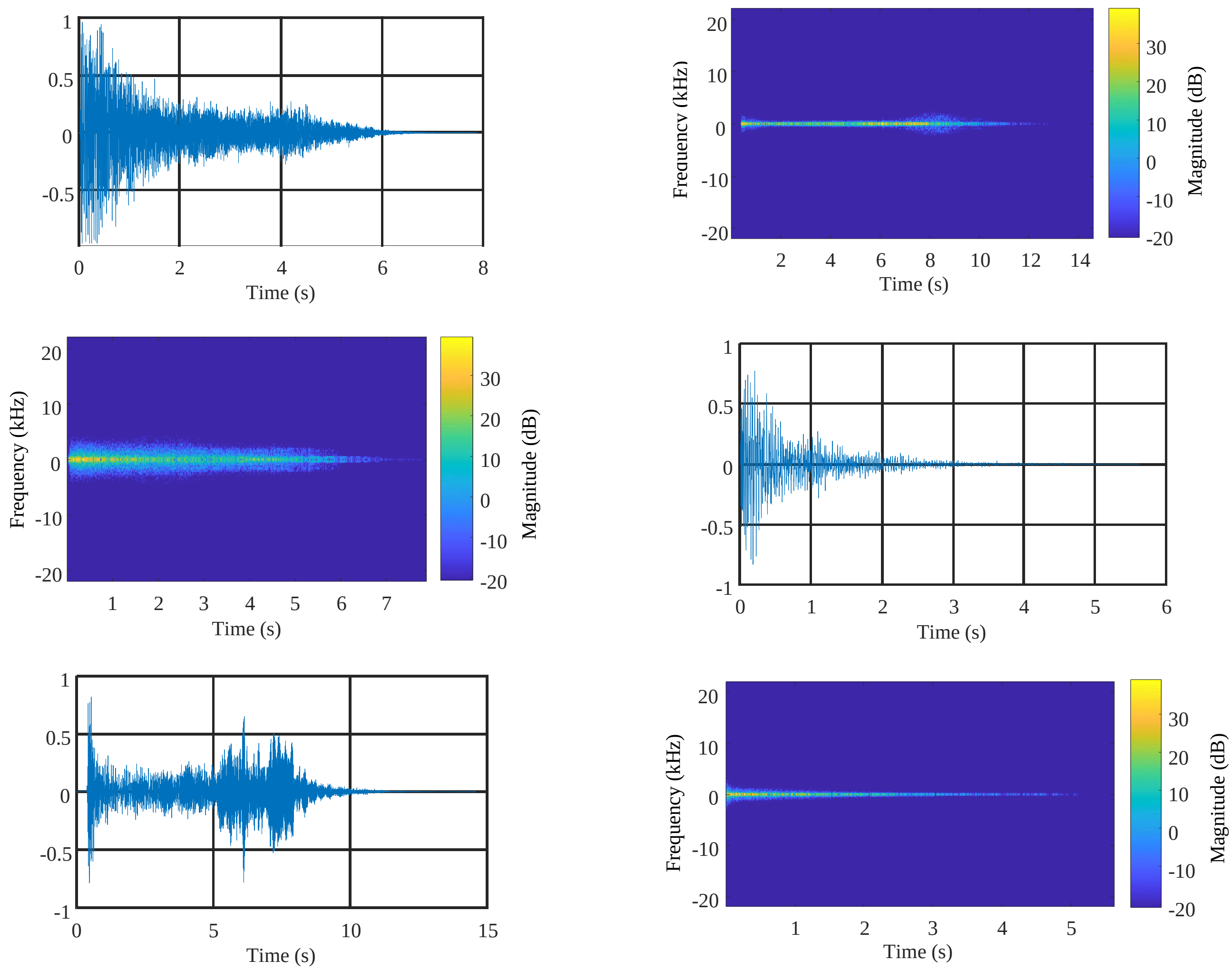


Fig. 4 GAN – audio synthesis



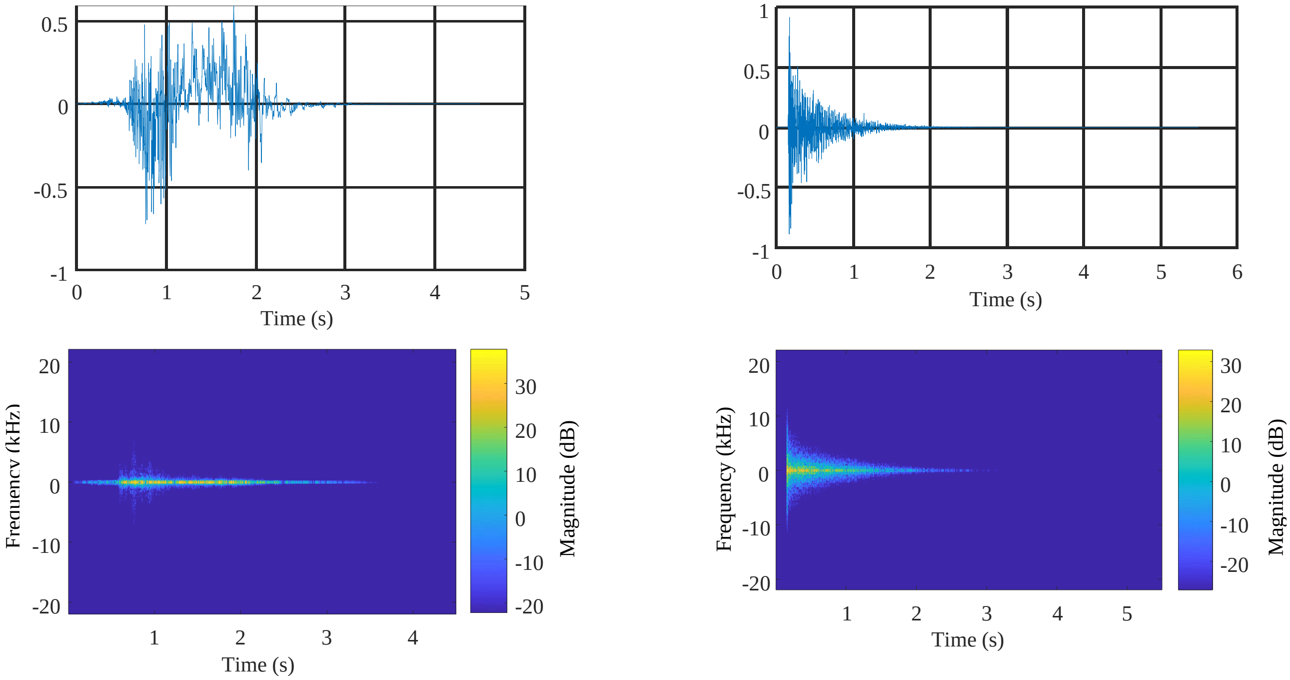


Fig. 5 Generated drum beats using GAN

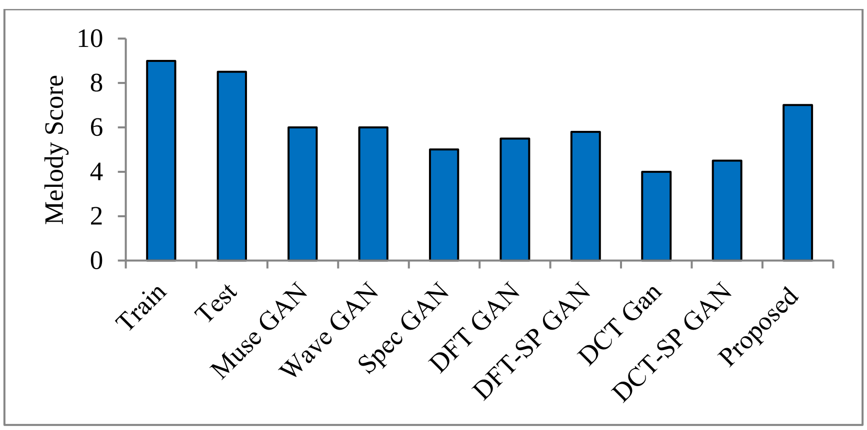


Fig. 6 Subjective evaluation: melody score

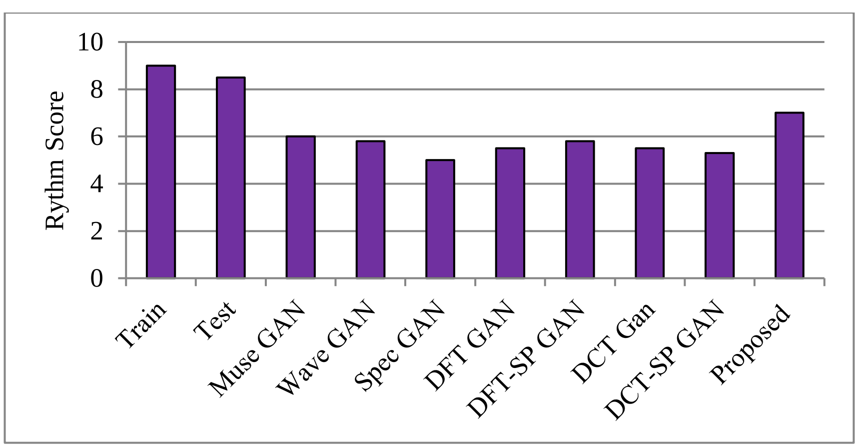


Fig. 7 Subjective evaluation: rhythm score

5.1. Subjective Assessment

A subjective analysis of the synthesized music files used for training was carried out. For each model discussed here, 10 sample music files were generated, various subjects analyzed the models, and the scores based on melody (Figure 6) and rhythm (Figure 7) were given. The proposed method based on GAN and STFT works on par with the existing audio synthesis algorithms**.**

6. Conclusion

Thus, GAN-based drum beat generation has proven to be an efficient way of auto-composing music files. In this work, the authors attempted to generate synthetic drum beats by training the GAN with the STFTs of the music file rather than the original music file itself.

From the synthesized music file, it is clear that the implemented methodology of music generation works well for drum beat synthesis using GAN. Though a quantitative assessment of the synthetic drum beats was not performed, the subjective evaluation by various audiences is acceptable. The author looks forward to scaling this to real-time music generation with multiple tracks.

References

[1] Maximos A. Kaliakatsos–Papakostas, Andreas Floros, and Michael N. Vrahatis, “evoDrummer: Deriving Rhythmic Patterns through Interactive Genetic Algorithms,’ *Lecture Notes in Computer Science, Springer,* vol. 7834, pp. 25-36, 2013. [CrossRef]

[2] Cárthach Ó Nuanáin, Perfecto Herrera, and Sergi Jordà, “Target-Based Rhythmic Pattern Generation and Variation with Genetic Algorithms,” *Sound Music Computing Conference*, 2015.

[3] Jean-Pierre Briot, Gaëtan Hadjeres, and François-David Pachet, *Deep Learning Techniques for Music Generation,* p. 284, 2020. [CrossRef]

[4] Ian J. Goodfellow et al., Generative Adversarial Nets, *Advances in Neural Information Processing Systems*, vol. 27, 2014.

[5] Tero Karras, Samuli Laine, and Timo Aila, “A Style-Based Generator Architecture for Generative Adversarial Networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 12, pp. 4217-4228, 2020. [CrossRef]

[6] Lantao Yu et al., “SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient,” *31st AAAI Conference Artificial Intelligence, AAAI 2017,* pp. 2852–2858, 2016.

[7] Gabriel Lima Guimaraes et al., “Objective-Reinforced Generative Adversarial Networks (ORGAN) for Sequence Generation Models,” 2017. [CrossRef]

[8] Hao-Wen Dong et al., “MuseGAN: Multi-Track Sequential Generative Adversarial Networks for Symbolic Music Generation and Accompaniment,” *The Association for the Advance of Artificial Intelligence,* 2018.

[9] R. Tuğrul Erdem et al., "Impact Analysis of a Concrete Beam via Generative Adversarial Networks," *International Journal of Recent Engineering Science*, vol. 9, no. 1, pp. 16-21, 2022. [CrossRef]

[10] Gargi Sharma, and Gourav Shrivastava, "Crop Disease Prediction using Deep Learning Techniques - A Review," *SSRG International Journal of Computer Science and Engineering*, vol. 9, no. 4, pp. 23-28, 2022. [CrossRef]

[11] Vrinda. K, and Dhanesh G. Kurup, "A Broadband Equivalent Model of On-Chip Spiral Inductors using Differential Evolution Algorithm," *International Journal of Engineering Trends and Technology*, vol. 70, no. 3, pp. 249-253, 2022. [CrossRef]

[12] Andrew Shin et al., “Melody Generation for Pop Music via Word Representation of Musical Properties,” arXivpreprint arXiv:1710.11549, 2017. [CrossRef]

[13] Hao-Wen Dong et al., “Musegan: Demonstration of a Convolutional GAN based Model for Generating Multi-Track Piano-Rolls,” *Late Breaking/ Demos, 18th International Society for Music Information Retrieval Conference,* 2017.

[14] Li-Chia Yang, Szu-Yu Chou, and Yi-Hsuan Yang, “Midinet: A Convolutional Generative Adversarial Network for Symbolic Domain Music Generation,” arXiv preprint arXiv:1703.10847, 2017. [CrossRef]

[15] Chris Donahue, Julian McAuley, and Miller Puckette, “Synthesizing Audio with Generative Adversarial Networks,” arXiv preprint arXiv:1802.04208, 2018.

[16] Narain Krishnamurthy, Beatgan – Generating Drum Loops via GANS, 2018.

[17] Magenta. [Online]. Available: https://magenta.tensorflow.org

[18] DeepJazz. [Online]. Available: https://deepjazz.io

[19] BachBot. [Online]. Available: https://bachbot.com

[20] FlowMachines. [Online]. Available: https://www.flow-machines.com

[21] WaveNet. [Online]. Available: https://deepmind.com

[22] Satya Srinivas Maddipati, and Malladi Srinivas, "Efficient Dimensionality Reduction using Improved Fuzzy C-Means Entropy Approach with Caps-TripleGAN for Predicting Software Defect in Imbalanced Dataset," *International Journal of Engineering Trends and Technology,* vol. 70, no. 7, pp. 1-9, 2022. [CrossRef]

[23] Radford, L. Metz, and S. Chintala, “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks,” *4th International Conference Learning Represent, ICLR*, 2016.

[24] Chris Donahue, Julian McAuley, and Miller Puckette, “Adversarial Audio Synthesis,” *In Proceedings of ICLR*, 2019.
[25] T. Kujani, and V. Dhilip Kumar, "Emotion Understanding from Facial Expressions using Stacked Generative Adversarial Network (GAN) and Deep Convolution Neural Network (DCNN)," *International Journal of Engineering Trends and Technology*, vol. 70, no. 10, pp. 98-110, 2022. [CrossRef]
[26] Suman Maria Tony, and S Sasikumar, “Music Generation Using Supervised Learning and LSTM,” *Advances in Energy Technology, Lecture Notes in Electrical Engineering, Springer*, Singapore, vol. 766, 2021. [CrossRef]
[27] Vinayak K. Bairagi, Sarang L. Joshi, and Vastav Bharambe, "Implementing Concatenative Text-To-Speech Synthesis System for Marathi Language using Python," *SSRG International Journal of Electrical and Electronics Engineering*, vol. 9, no. 9, pp. 1-11, 2022. [CrossRef]
[28] Garv Agarwal, Sai Sanjeet, and Bibhu Datta Sahoo, "Seizure Prediction using Generative Adversarial Networks for EEG Data Synthesis," *SSRG International Journal of Computer Science and Engineering*, vol. 9, no. 10, pp. 1-7, 2022. [CrossRef]
[29] Nipun Agarwala, Yuki Inoue, and Axel Sly, “*Music Composition using Recurrent Neural Networks*,” Stanford University, Technical Report in CS224, 2017.
[30] Gaëtan Hadjeres, François Pachet, and Frank Nielsen, “DeepBach: A Steerable Model for Bach Chorales Generation,” *International Conference on Machine Learning*, New York City, NY, USA, pp. 1362-1371, 2016.
[31] Keerti Maithil, and Tasneem Bano Rehman, "Urban Remote Sensing Image Segmentation using Dense U-Net+," *SSRG International Journal of Computer Science and Engineering*, vol. 9, no. 3, pp. 21-28, 2022. [CrossRef]
[32] Huang KC, Jung Q, and Lu J, “*Algorithmic Music Composition Using Recurrent Neural Networking*,” Stanford University, Technical Report in CS22, 2017.
[33] T V Divya, and Barnali Gupta Banik, "An Integrated Cycle GAN and PEGASUS to Generate Synthetic Data for Detection of Fake News," *International Journal of Engineering Trends and Technology*, vol. 70, no. 8, pp. 57-70, 2022. [CrossRef]
[34] Suman Maria Tony, and S. Sasikumar, “Generative Adversarial Network for Music Generation,” *High Performance Computing and Networking. Lecture Notes in Electrical Engineering, Springer*, Singapore, vol. 853, 2022. [CrossRef]