Application of Deep Neural Networks to Music Composition Based on MIDI Datasets and Graphical Representation

Mateusz Modrzejewski, Mateusz Dorobek, Przemysław Rokita

Division of Computer Graphics, Institute of Computer Science, The Faculty of Electronics and Information Technology, Warsaw University of Technology, Nowowiejska 15/19, 00-665 Warsaw, Poland
M.Modrzejewski@ii.pw.edu.pl, mdorobek@mion.elka.pw.edu.pl, P.Rokita@ii.pw.edu.pl

Abstract. In this paper we have presented a method for composing and generating short musical phrases using a deep convolutional generative adversarial network (DCGAN). We have used a dataset of classical and jazz music MIDI recordings in order to train the network. Our approach introduces translating the MIDI data into graphical images in a piano roll format suitable for the DCGAN, using the RGB channels as additional information carriers for improved performance. We show that the network has learned to generate images that are indistinguishable from the input data and, when translated back to MIDI and played back, include several musically interesting rhythmic and harmonic structures. The results of the conducted experiments are described and discussed, with conclusions for further work and a short comparison with selected existing solutions.

Keywords: AI, artificial intelligence, neural networks, GAN, music, MIDI

1 Introduction

Music is a vital part of our lives - it's a deeply human phenomenon that has emerged in some form in every civilization throughout history. Music as we know and perceive it today has started developing in the late XVIII century, but its beginnings reach out as far as to 3000 BC [6]. Music is constantly evolving and one hand, we may consider its evolution only in terms of the skills and science behind the brilliance of master instrumentalists, composers, singers, producers and songwriters. However, the second obvious factor of the evolution of music is deeply rooted in technological progress - from instruments such as synthesizers, through digital audio workstations (DAW) up to CPU-consuming algorithms that emulate vintage devices or help us build microtuning systems. The evolution of music is also closely related to achievements such as the magnetic tape, the compact disc and improvements in amplification systems, just to name a few examples.

Using artificial intelligence for generating music is interesting both in scientific and artistic terms:

- due to the abstract character and overall complexity of music, it is a very challenging information type for AI solutions;
- it expands our overall knowledge of how we perceive music and allows us to investigate the details of our creative process;
- as an undiscovered and highly experimental form of composition, it may be used to greatly stimulate the artist's creativity and offer new, unconventional forms of expression.

Besides of purely artistic applications, the need of such solutions is clearly visible with the development of such environments like Google Magenta [9] and attempts on music classification [1] and generation [7] [3], as well as with the expansion of the content creation market in modern media.

We propose to generate music using a graphic representation. We have created a dataset of images by transforming a quantized MIDI recordings dataset consisting of a classical music part and a jazz music part. The images were used for training the neural network to generate similar ones, which after decoding back to MIDI could be listened back to. Our expectation was therefore to train a network to generate samples capturing the overall character, as well as a certain level of harmonic and rhythmic content that could be found in the training data.

2 Datasets

2.1 Our approach to data representation

MIDI files follow a protocol in which subsequent lines represent key and control actions. It is stored in a binary form and can be converted into a text format [11]. Although MIDI data can be used in the text format, we propose to use a latent graphical representation for the data. Our approach is to use an enhanced piano roll graphic format [12]. Piano roll represents both the time structure of music (rhythm), as well as the harmonic and melodic structure (pitches) in a comprehensible format. Our data processsing consists of:

- redundant information (comments etc.) is removed from the MIDI text,
- the dynamics of the samples are all set to maximum,
- all the samples are quantized to 30ms (16th note in 120BPM tempo),
- redundant long pauses are removed.

The MIDI data is then compressed to 64x64 images as shown on Fig. 1.

The scale of the piano has been reduced to 64 keys from the actual 88 by scaling the far notes by an octave, as the loss of information from these notes does not introduce particular modifications in the overall character of the samples.

The rhythm structure is also compressed by using the RGB channels as additional information carriers: each of the notes is coded using all of the subpixels of the bitmap, therefore stretching the timeline three times. Rhythm values are

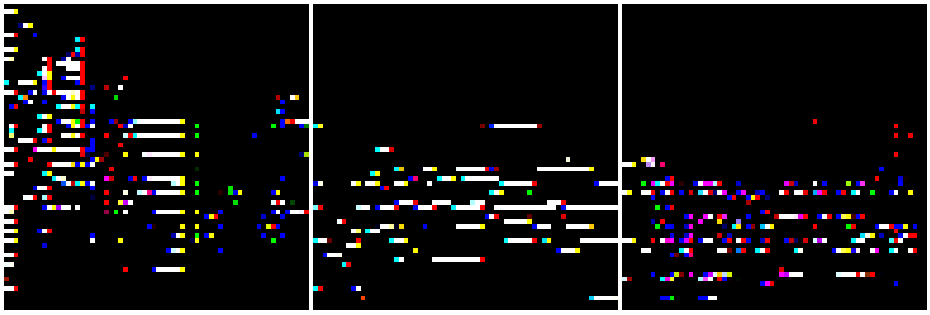


Fig. 1. Example MIDI files coded as RGB images, used for training the network.

represented by lighting up the subpixels: this allows to represent 20 seconds of music with each of the R, G, B, C, M, Y, K and W colored images, which is a sample length better than in most available solutions.

In some of the used training files, we have encountered an error as a result of incorrectly closed sustain pedal[1] events. The images containing this error had long white lines followed by an abrupt stop, as shown in Fig 2. We have decided not to remove these images, but rather to automatically shut the sustain pedal after 3 seconds (a time long enough for most of the actual situations where a piano player would use the pedal in the considered music examples). This additional processing stage also improved the rhythmic clarity of some of the data, while not introducing significant changes to the overall musical character of the data.

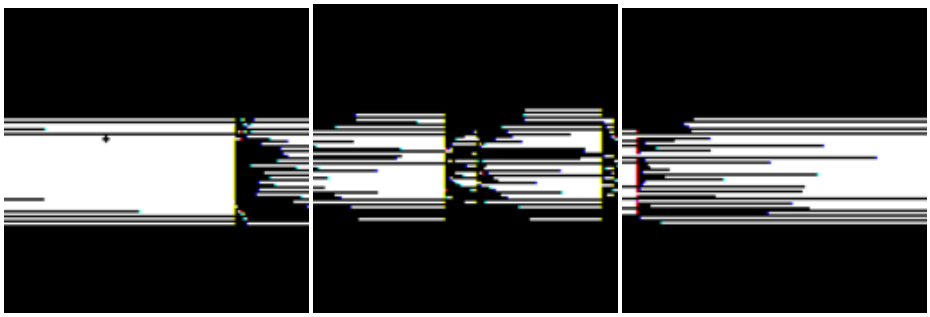


Fig. 2. Example of the error found in some data.

[1] While the sustain pedal is pressed, the notes of the piano sustain after keys are released by the pianist.

2.2 Qualities of selected datasets

Due to a general lack of usable data in the piano roll format, we have decided to transform the MAESTRO MIDI dataset[5] into piano roll images. MAESTRO consists of 172 hours of virtuoso solo piano classical music in MIDI and WAV format. The files were collected from recordings of different piano players from a piano competition in Minneapolis, USA. We have also used an over 20 hours MIDI dataset of jazz pianist Doug McKenzie's recordings [2]. Table 1 shows a brief comparison of the qualities of the datasets.

Table 1. Features of the samples in the used data sets.

feature	**MAESTRO MIDI** (classical music)	**Doug McKenzie dataset** (jazz piano)
number of samples	25k images (ca. 172 hrs)	2,9k images (ca. 20hrs)
overall level of musical technical difficulty	high	high
overall rhythmic structure	presence of many non-repetive phrases with fluent tempo changes typical for classical music	varying, improvised rhythmic structures, presence of triplet phrasing in a swing context
harmonic structure	ordered, often very typical for the rules of classical music	rich, includes more complex harmonies typical for jazz music (some of which are improvised and some are well-known, typical jazz chord progressions)
dynamic structure	full range of dynamic (from very soft to very loud)	full range of dynamic
instruments	solo piano	mostly solo piano, some files with double bass and drums

3 Method and summary of our approach

We have used a DCGAN implementation using PyTorch with the model structure as described in [4]. All experiments were performed using a Nvidia GPU with CUDA architecture. DCGAN contains two concurring convolutional networks: the *generator*, which is trying to create fake images similar to real ones, to fool the *discriminator*, which has to distinguish the training images

from the generated images. Both of them have a CNN [13] structure to analyze and extract features from 2D matrices.

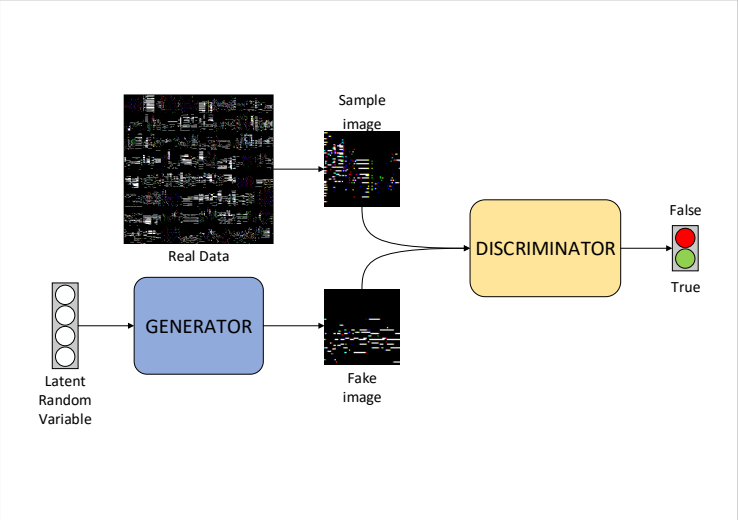


Fig. 3. Structure of the DCGAN.

All experiments were conducted used the training images that we have created from our MIDI datasets.

Upon performing qualitative experiments with the environment proposed above, we have also decided to perform additional experiments with an expanded dynamic range. In our main experiment, we have used flat dynamics set at the maximum value - the full dynamic range experiment was conducted in order to investigate how the dynamic spectrum affects the training process.

4 Results

4.1 MAESTRO dataset

In first experiment we have used generated images with binary dynamic range (on or off) generated from MAESTRO database. We have trained the network for 50 000 iterations, also trying whether an additional 20 000 iterations would improve the results.

Figure 4 shows the flat dynamic results. At first glance we can't distinguish real images from the fake ones, as the included structures are very similar. Figure 6 shows the loss functions for the two experiments. Additionally, Figure 5 shows single sample result images both without and with full dynamics.

Upon translation back to MIDI and listening through the generated material, we have found the presence of the following musical composition elements:

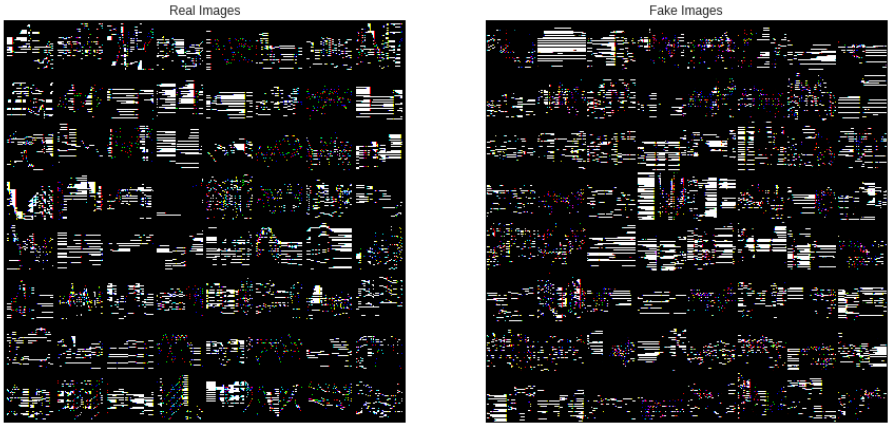


Fig. 4. Results generated with the MAESTRO dataset.

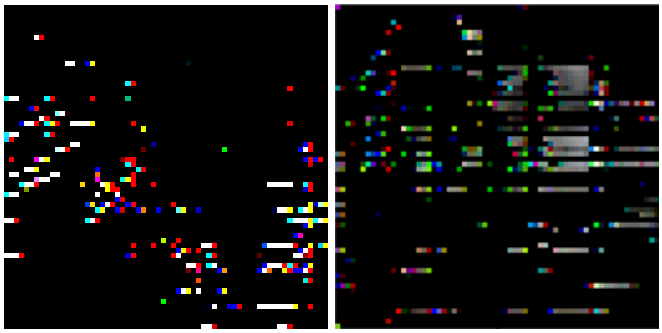


Fig. 5. Example result images with flat (left) and full (right) dynamic with the MAESTRO set.

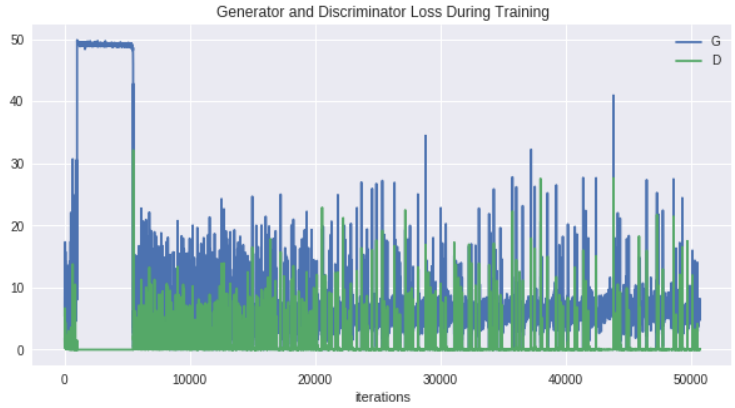
- major and minor chords;
- typical voicings and classical music cadenza resolutions;
- 4th chords;
- pronounced bass lines and arpeggios;
- V-I chord progressions (dominant to tonic chord - one of the most important chord progressions in music theory) or a tonic chord at the end of a phrase;

The generated music had mostly a quite chaotic rhythmic structure, but that is due to the dynamic quantization and training using virtuoso performances, often including very fast and difficult pieces. Some of the samples have a much more pronounced and deliberate rhythmic structure, with clear phrases built out of eight and sixteenth notes.

Unfortunately, no significant improvement was introduced in the results after running 20 000 additional iterations, as much of the chaotic character remained unchanged. The harmonic elements were also similar to the previous experiment.

The generator's cost is oscillating around a certain value, while the discriminator almost perfectly recognizes false images from real ones, as its loss function is close to 0.

a)



b)

Fig. 6. Loss function for a) MAESTRO dataset b) MAESTRO dataset with additional iterations.

As we can see in Figure 7, in the full dynamic experiment the discriminator cost went up and generator cost fell down to zero ofter 25 000 iterations, which means that the discriminator couldn't tell any difference between fake and real images. As suspected, the results were more chaotic both rhythmically and harmonically.

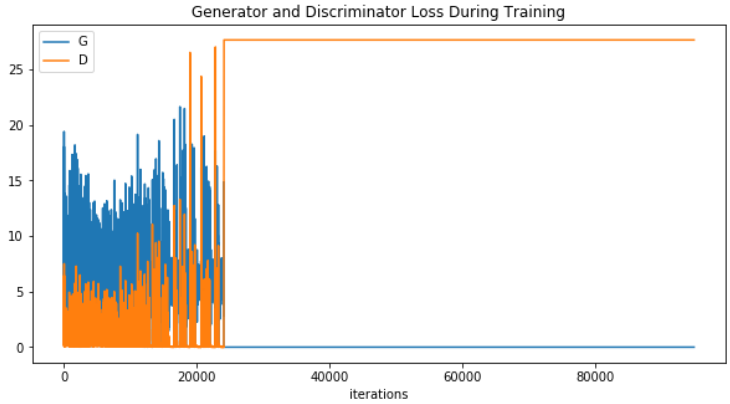


Fig. 7. Cost function in the full dynamic experiment.

4.2 Doug McKenzie MIDI dataset

In this experiment we have tried to receive similar results as in first one, but using a jazz music database. A 10 times smaller database allowed us to perform 120 000 iterations in approximately the same time as previous experiments. Figure 8 shows that after 100 000 iterations the generator cost went up, and discriminator cost fell down to zero - we can observe a learning reversal. In Figure 9 we can observe difference between images generated before and after the drastic generator const increase.

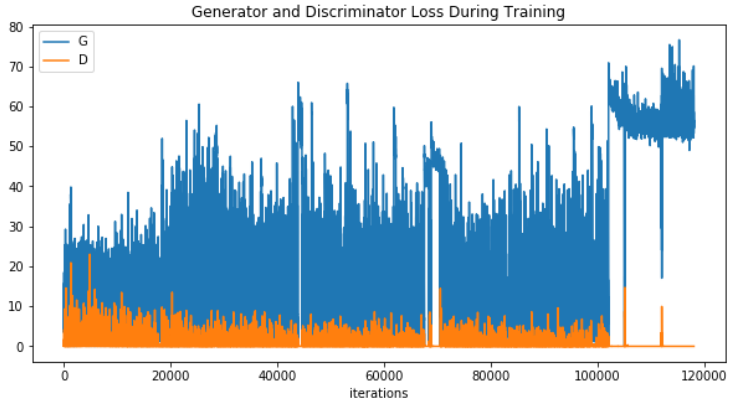


Fig. 8. Loss function with smaller database.

The results after the reversal were obviously a dense cluster of notes lacking major musical sense, but listening back to the results of iterations directly pre-

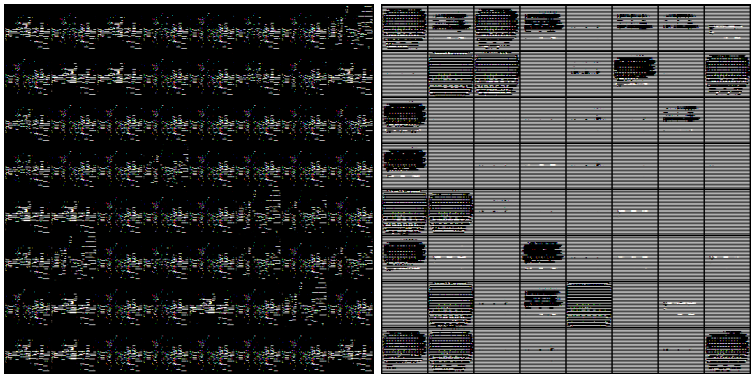


Fig. 9. Sample results obtained before (left) and after (right) learning reversal.

ceding the learning reversal we have observed satisfying results, including the presence of:

- altered chords typical for jazz and blues music;
- phrases finished by complex altered chords;

in addition to the elements mentioned above with classical music that can also be found in jazz music.

4.3 Result summary

Generated images in most cases were indistinguishable at first glance for a human (and in the experiment with full dynamic range even to discriminator). We are able to generate 20 second phrases with 64x64 images and 40 second phrases if dropping the piano keyboard reduction (but also largely increasing the training time). Most of our results have advanced harmonic structures, but quite chaotic rhythm. Melodic and bass lines showed up several times and so did arpeggios and even cadenzas with typical resolutions. Some examples also contained advanced jazz chords and tonic chords at the end of the phrase. Existing solutions for a similar problem of generating music (such as Magenta, DeepJazz [8] or Amper Music [10]) create music phrases that are much simpler and shorter, with just basic harmony and certain pre-defined or overfitted solutions. Most of them have a regular rhythm that is less chaotic than in our approach, but music generated by our DCGAN contains advanced progression with resolutions which is our main advantage over other similar projects.

5 Conclusions and further work

In this paper we have proposed a method for composing short musical phrases using a deep convolutional generative-adversarial network and a graphic representation of MIDI input data. The samples generated by our solution are longer

and have a richer harmonic structure when compared to results generated by many of the existing solutions. We have selected a set of musical qualitative features (harmony, rhythmic structure etc.) that our network has learned to reproduce. We have also performed additional experiments in order to determine parameters allowing for overall improvement of the quality of the generated samples.

The obtained results allow us to conclude that a GAN and a graphical piano roll data representation for, however unorthodox, is a good choice for further experiments with generating music. In our nearest work we would like to focus on generating long, cohesive musical ideas and genre-specific harmonic and rhythmic content.

References

1. Keunwoo Choi, Gyorgy Fazekas, Mark Sandler, and Kyunghyun Cho. Convolutional recurrent neural networks for music classification. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2392–2396. IEEE, 2017.
2. `https://bushgrafts.com/midi/` D. McKenzie. *MIDI Collection*, Online, access 16 Jan 2019.
3. Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Douglas Eck, Karen Simonyan, and Mohammad Norouzi. Neural audio synthesis of musical notes with wavenet autoencoders. In *ICML*, 2017.
4. Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.
5. Curtis Hawthorne, Andriy Stasyuk, Adam Roberts, Ian Simon, Cheng-Zhi Anna Huang, Sander Dieleman, Erich Elsen, Jesse Engel, and Douglas Eck. Enabling Factorized Piano Music Modeling and Generation with the MAESTRO Dataset, 2018.
6. K.Wilkowska-Chołmińska J.Chołmiński. *Historia Muzyki cz.I.* PWM, 1989.
7. Daniel D. Johnson, Robert M. Keller, and Nicholas Weintraut. Learning to create jazz melodies using a product of experts. In *ICCC*, 2017.
8. `https://github.com/jisungk/deepjazz`. *DeepJazz*, Online, access 16 Jan 2019.
9. `https://magenta.tensorflow.org`. *Google Magenta - research tool for artistic applications of machine learning*, Online, access 16 Jan 2019.
10. `https://www.ampermusic.com/`. *Amper Music*, Online, access 16 Jan 2019.
11. `https://www.csie.ntu.edu.tw/~r92092/ref/midi/`. *The MIDI File Format*, Online, access 16 Jan 2019.
12. `http://www.pianola.co.nz/public/index.php/web/about_piano_rolls`. *What is a Player Piano (Pianola)?*, Online, access 16 Jan 2019.
13. Aaron van den Oord, Nal Kalchbrenner, Lasse Espeholt, koray kavukcuoglu, Oriol Vinyals, and Alex Graves. Conditional image generation with PixelCNN decoders. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 4790–4798. Curran Associates, Inc., 2016.