Deep Learning:

First Milestone

Problem summary:

We would like to make a music generator AI as our homework. Our idea is to try and convert our music-data into pictures and then use a GAN to produce picture, which the discriminator network cannot distinguish from real data based pictures. When we reach this level of performance we translate the pictures back to .midi, play them and hopefully we end up with listenable music.

Data acquisition, description:

For our data-set we had two criteria in mind: we needed .midi files and we wanted to use tracks similar to each other in style. The .midi files allow us to work with large data-sets (25-100GB as uncompressed .wav files) without actually having to store them as .wavs thus taking up only minimal storage capacity (not counting the preprocessed files). Them being similar in style is not necessary, however we thought that it would be best for our first approach to only consider tracks from the same style. We are planning to introduce other styles to the network in the future, so the data-set extensibility can also be considered a criteria. Luckily we found a well-made data-set called Magenta: Maestro[1], that satisfies all our needs. Maestro is a dataset composed of over 200 hours of virtuosic piano performances.

Data exploration:

Our data has the following main features:

- Time: Absolute time, in terms of MIDI clocks, at which this event occurs. Meta-events for which time is not meaningful (for example, song title, copyright information, etc.) have an absolute time of 0.
- Type/Event: Name identifying the type of the record. Record types are text consisting of upper and lower case letters and the underscore ("_"), contain no embedded spaces, and are not enclosed in quotes.
- Channel: The channel identifier.
- Note: The currently played note (integer 21-109, a piano has 88 keys)
- Velocity: The matching note's volume at the moment.

Further details at midicsv's website[2]

---

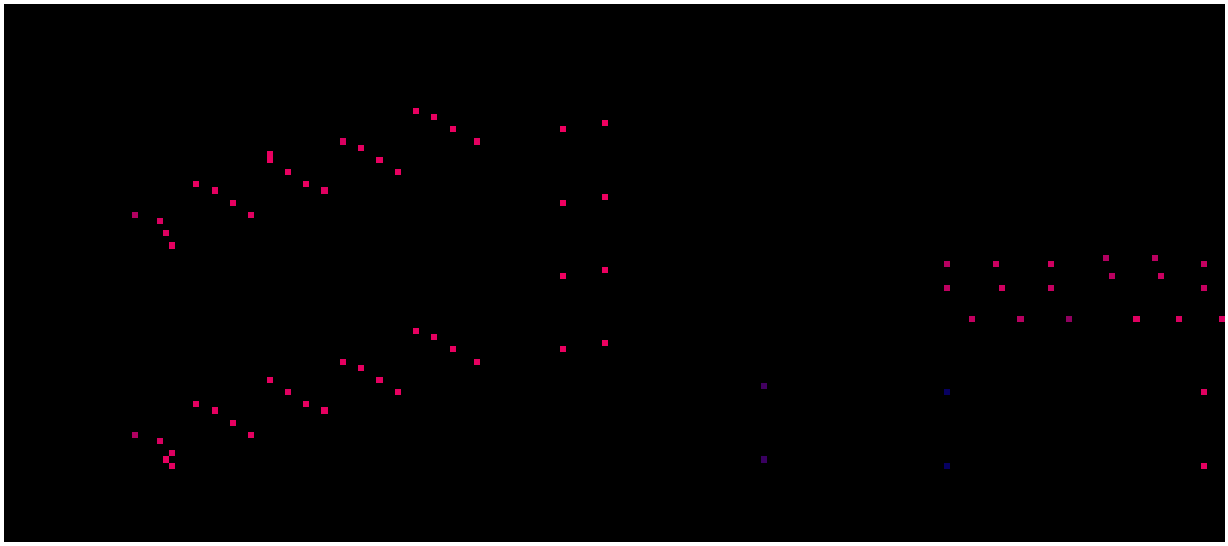
[1] Curtis Hawthorne, Andriy Stasyuk, Adam Roberts, Ian Simon, Cheng-Zhi Anna Huang, Sander Dieleman, Erich Elsen, Jesse Engel, and Douglas Eck. "Enabling Factorized Piano Music Modeling and Generation with the MAESTRO Dataset." In International Conference on Learning Representations, 2019. https://arxiv.org/abs/1810.12247.

[2] "MIDICSV: Convert MIDI File to and from CSV - Fourmilab." https://www.fourmilab.ch/webtools/midicsv/..

Data preprocessing:

Here we only explain the main steps and ideas of our preprocessing, because we include a well-commented source code, that produces our actual training (70%), validation (20%) and test (10%) data.
First we read the .midi files and transform them to only contain the relevant informations we need (notes length, instrument type, tempo, volume), then we transform them into 10 sec long parts (if the music is not a multiple of 10 sec long, then we “add silence” to the end).
The fixed time long parts can then be transformed into pictures, where the RGBA channels are the notes length, the instrument type, the tempo and the volume respectively. The pixels height indicates the note itself.

An example for the resulting pictures:



In our first approach this will be the way we process the data, but we would like to note here, that we are thinking about different approaches as well. Another promising way can be the usage of Hilbert’s curves. In a nuttshell: Hilbert’s curves[3] is a way to map the 1D-like spectrogram to a 2D picture, in such way, that the notes close to each other on the spectrogram will also be close on the 2D picture. This approach might work better with convolutional networks since the small matrices, parsing the picture can actually extract features that correlates more.

[3] "3Blue1Brown - YouTube." https://www.youtube.com/channel/UCYO_jab_esuFRV4b17AJtAw.