Growing Sounds: Spectrogram Generation with Neural Cellular Automata and Synthesis with BigVGAN

Milo Beuzeval
Music Technology Group
Universitat Pompeu Fabra
Barcelona, Spain

***Abstract*—Neural Cellular automata have shown to be good at growing visual textures and patterns. However their application to spectrogram generation remains, as far as we are aware, unexplored. In this work, we investigate the challenges of adapting NCA to the audio domain by training models to generate Mel-spectrograms that attempt to resemble target sounds. We experiment with loss functions that incorporate the features from pre-trained VGG and VGGish networks to try encourage structured growth, as well as additional perception filters. Our findings provide insights into the challenges of combining NCA-based texture synthesis with mechanisms to learn meaningful audio representations and so we suggest future directions for improving our implementation via aligned preprocessing between the neural vocoder (BigVGAN) and VGGish model. In addition to more sonically motivated convolutional filters.**

I. Introduction

The generative capacity of CA exhibit from simple rules to higher level complex and often repeatable patterns is a remarkable and useful trait for generating new patterns of sound particularly for generative music compositions and has been used frequently [1].

Cellular automata have also been used successfully for synthesising sound directly via the control of granular synthesis [2]. However in comparison its use in generative composition, there has been less work in directly constructing the sound wave or spectrogram with cellular automata directly.

The recent of advances in neural cellular automata (NCA) have provided some exciting results [3], [4]. NCA differ from traditional hard-coded rule CA, in that the update rules are differentiable and thus learnable via deep learning frameworks. Whilst these works mainly focused on investigating toy models for exploring biological phenomena such as morphogenesis and regeneration of patterns. Some of the overarching capabilities shown was a way of training a model that learned a kind of malleable, persistent representation of the target image it was trying to learn. These properties are particularly appealing characteristics one might want when trying to model a sound and generate new, potentially surprising, but similar sounds.

Neural vocoders like BigVGAN [5] have significantly improved the quality of spectrogram-based audio synthesis compared to traditional approaches like Griffin-Lim [6]. While Griffin-Lim reconstructs phase iteratively and often introduces artifacts, neural vocoders learn a direct mapping from spectrograms to waveforms, producing more natural and high-fidelity audio with generally better temporal coherence. This advancement makes them quite powerful and suitable for our requirements in reliably synthesizing spectrograms.

With these ideas in mind, we propose building on this work using NCA to "grow" Mel-spectrograms with which we can synthesise into audio with a pre-trained BigVGAN network.

II. Literature review

Existing work using CA for sound has mainly focused on using the patterns it generated for compositions rather than direct synthesis of audio. Such as the liquiPrism which uses CA across the faces of a cube to trigger a synthesizer [7].

Serquera and Miranda explore sound synthesis using cellular automata, specifically the multitype voter model, which transforms two-dimensional cellular automaton evolutions into spectrograms [8]. Via histograms of the CA, their approach treats bins as potential sound partials, where each bin's changing value represents the amplitude of a different frequency component over time. They develop the techniques to control attack and release patterns, however overall found that they were relatively difficult to control.

There has been some recent interest more generally in adapting the ideas in the field of artificial-life for use in music generation. A good recent example of this is the recent work of Tölvera python package [9]. This provides a way of using basal intelligence - various artificial life algorithms such as slime mould simulations, cellular automata like systems and so on particularly for artistic applications such as generative music.

The features learned in the layers of discriminately trained convolutional neural networks (CNNs) have been shown to be useful when integrated within a loss function, fulfilling the role of a discriminator guiding a generative process towards an intended outcome based on the knowledge endowed in the CNN network. This has been shown to work well for visual textures [10].

III. Methodology

In this section we introduce our approach and how we attempt to grow mel-spectrograms with a NCA, then feed the generated image into the BigVGAN to produce a waveform.

We focus on explaining the key components of our implementation and particularly on the details where our approach differs from existing similar work [4] in the visual domain.

A. Neural Cellular Automata for Spectrogram Generation

We base our implementation of a NCA on the approach set out in [3], [4]. The starting point for our final implementation is based on the simple PyTorch example provided in the original paper [4]. After some initial experimentation with a the model described in [3] we found that the the model in [4] worked better for the moderately higher resolutions required to work with the BigVGAN neural vocoder.

1) NCA Implementation: The core of our system implements a NCA which maintains for each pixel in the spectrogram a vector of values. We vary the size of this somewhat in our experiments but we start with 16 real values. In our implementation, the first 3 can be used for the RGB values or just the first for a greyscale image, for the visual data displayed in the image. The rest are hidden channels that the update rule learns how to use for inter-cell communication.

The forward pass through the NCA consists of a perception layer, which is a set of small hard-coded convolutional filters, in order for cells to extract local spatial features from the cells around them. Identity, Sobel and Laplacian filters are used. We experiment with different filters later, but our base implementation uses these as done in the aforementioned original implementation [4]. Now that the perception has abstracted a feature representation of the image that encodes some information about local structure and gradients, this is passed into the first convolutional layer. Projecting them into a high dimensional space of 96 channels and through a ReLU. Then this is passed through a second convolutional layer that maps back to the initial state size. Finally, we apply an stochastic update mask that updates a randomly selected 50% of the cells. This asynchronicity in updates, is to simulate a more natural growth process.

2) Calculating Loss with VGG16 & VGGish: To calculate the loss of our network, we implement two approaches. The first is the approach found in [4] which pushes the output state from the NCA through a pre-trained VGG network, extracts the features at the non-linear activation layers and calculates the optimal transport loss between between the features of the current state and the features of the target image.

In our work we are interested specifically in Mel spectrograms and so instead of VGG which is trained on natural images from the ImageNet dataset, we would like to use something more specific to our chosen domain. For this, we select VGGish and use the pre-trained model available in the torch-vggish python package [**?**].

VGGish is trained on a large dataset of audio clips from the AudioSet dataset [11] and is specifically designed to process mel-spectrogram representations. Its architecture is very similar to that of the original VGG style networks, however unlike VGG which has learned patterns of natural images, the pre-trained VGGish model has learned representations that capture the temporal and spectral structures characteristic of audio signals. This we think makes it particularly well suited for our task, as it can extract meaningful features from mel-spectrograms, the NCA to be guided by a loss function tailored to the auditory domain. Due to the architectural differences, we select indices for the correct ReLU layers for the VGGish model in our new loss function.

3) Perception filter changes for audio: We incorporate additional filters into the perception layer of our NCA model so as to better suit the spectrogram domain. Whilst the original filters were mainly designed to mimic typical natural phenomena such as chemical gradients, we do not want to restrict our model in the same way.

Unlike natural images, mel-spectrograms encode time-frequency representations of audio, so we add filters which try to better capture the temporal and spectral structures present in this representation. We introduce several new filters which specifically target different aspects:

1) Horizontal filter: capture temporal patterns along the time axis
2) Vertical filters: capture patterns in frequency
3) Diagonal Filters: capture oblique patterns such as onsets
4) Broad context filter: provide a slightly larger receptive field

Along with the original filters (identity, Sobel filter for edge detection and second order derivatives in the Laplacian filter) the additional filters should provide an expanded set of features through which the NCA model can capture patterns specific better suited to the Mel-spectrogram compared to natural image.

B. Generating examples

C. Vocoding with BigVGAN

We use the *bigvgan_v2_44khz_128band_512x* pre-trained model provided by NVIDIA on hugging face. This is the highest quality version released. We install the BigVGAN python package from the repository and load the weights. The authors of BigVGAN python package with utilities with which you can generate a mel-spectrogram in tensor form and convert this to sound using a pre-trained model. We implement additional functionality to convert this tensor to an image and save it. We test reading back from mel-spectrograms saved as PNG files to waveforms using the BigVGAN network vs doing so directly from the mel-tensor and get very good quality sound for both methods. In both cases, there is some noticeable lack of punch to the onsets of a sound compared to the original audio file. However, this is inherently a problem with mel-spectrogram representation discarding phase information. Nonetheless, this level of quality is sufficient for our experiments, as our aim is not for perfect reconstruction but rather to learn a malleable representation that encourages possible variation from the target sound, and so a little more will not go amiss.

D. Difficulties with our current approach

The BigVGAN model and VGGish expect different pre-processing to be done to generate log Mel spectrograms.

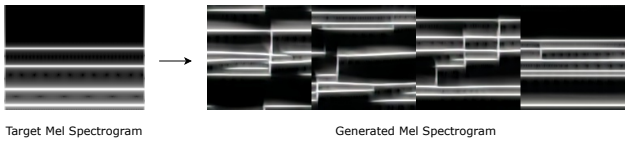

Fig. 1. A target mel spectrogram of some sine waves and the generated spectrograms from the NCA model + VGG feature sliced optimal transport loss model.

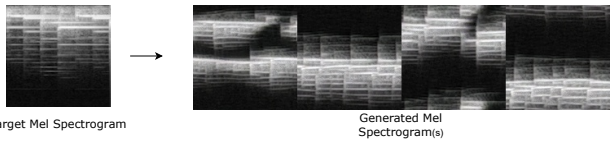

Fig. 2. A target mel spectrogram of a piano sound and the generated spectrograms from the NCA model + VGG feature sliced optimal transport loss model. Note that here the images are upside down, since this is the BigVGAN model mel-spectrograms are generated.

To be able to use both the pre-trained networks we have compromised and used the approach used by BigVGAN to generate our mel spectrogram. This limits the effectiveness of using the VGGish features however, when training the NCA VGGish model we still see the loss decrease.

IV. Results

A. Results with VGG

1) Target: Simple sine wave pattern: Our first target Mel spectrogram consists of a simple combination of sine waves at different frequencies. This experiment aimed to determine whether the model could converge to a simple structured pattern like this or if it would struggle to maintain the correct spectral representation. As shown in Figure 1, the trained NCA model captures some of the visual characteristics of the Mel spectrogram, producing distinct solid lines. However, these lines are not consistently placed at the correct frequencies, and additional structures emerge that are absent in the target, such as a staggered, stair-like diagonal connections between the lines.

While the model successfully learns certain visual components of the target, it does not develop a representation that aligns with the underlying audio properties. The presence of white frequency bands across the image visually give the impression of an accurate reconstruction, but these do not correspond meaningfully to the frequency content required for the later synthesized sound to match the target. This suggests that the model is not sensitive to the perceptual aspects of the audio domain, leading to outputs that look visually similar in some respects but do not produce the intended sound when converted back to audio. We refer the reader to the accompanying demonstration website where they can hear an example of this.

2) Target: Piano sound: We test a second target Mel spectrogram, this time from a simple piano sound. As with the previous sine wave experiment, the model captures some key visual characteristics, such as the presence of harmonics and their relative intensity. When re-synthesizing the output Mel spectrograms using BigVGAN, we hear some resemblance to the original sound, particularly in the percussive clunks of key presses between notes. However, the overall timbre of the piano is not well preserved; the output sounds fuzzy and synthetic rather than clear and natural like the original sound.

Interestingly, when generating from noise with the trained model, the pattern produced by the trained NCA once "converged" does not remain static over the following steps, as was the case for the simple sine wave pattern. Instead, the generated structure shifts downward over time, creating a vertical drift in the spectrogram. This effect likely arises due to the way NCAs update their state: local update rules propagate patterns iteratively, and without an explicit mechanism to anchor features in place, structures can gradually drift as the model evolves its output. While this movement is inaccurate in terms of accurately replicating the target, it highlights an emergent behaviour unique to the NCAs self-organizing nature, suggesting both its limitations and its potential for creative exploration in this domain.

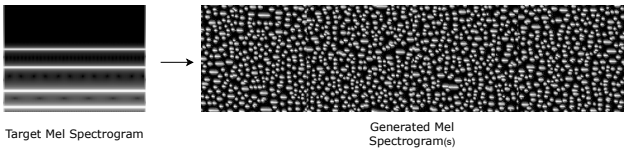

Fig. 3. A target mel spectrogram of some sine waves and the generated spectrograms from the NCA model + VGGish feature sliced optimal transport loss model.

B. Results with VGGish

1) Target: Simple sine wave pattern: We repeat the sine wave experiment using our updated model, which incorporates the VGGish network to compute the loss. The results for a new target are shown in Figure 3. In the generated spectrograms, we observe mainly dotted points appearing, with some signs of horizontal clustering, perhaps suggesting an attempt to form continuous frequency bands similar to those in the target. However, the overall structure of the generated patterns do not capture the expected frequency organization of the target.

We suspect these poor results stem from a combination of factors. Firstly, our target Mel spectrograms are pre-processed differently from what the VGGish network expects, potentially making the extracted features less meaningful for guiding learning. Secondly, our current learning rate schedule may not be optimal. While the training loss consistently decreases, we hypothesize that starting with a higher learning rate and gradually annealing it could encourage better convergence toward a pattern that more closely resembles the target.

2) Target: Clicking sound: We next evaluate our VGGish-based model using a new target: the percussive clicking sound of a stove ignition. This test allows us to examine how the model handles a different type of pattern in the Mel spectrogram, one characterized by strong vertical bands, typical of transient sounds with sharp onsets. Additionally,

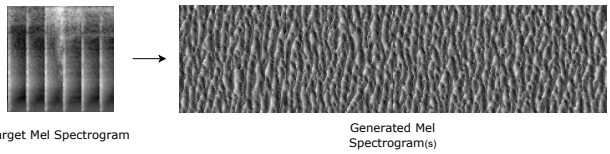

Fig. 4. A target mel spectrogram of a stove ignition clicking and the generated spectrograms from the NCA model + VGGish feature sliced optimal transport loss model.

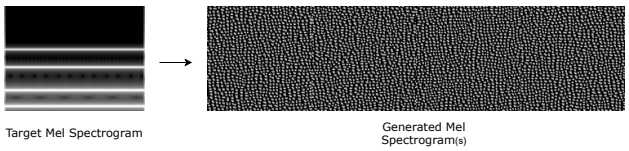

Fig. 5. A target mel spectrogram of some sine waves and the generated spectrograms from the NCA model with our extra perception filters + VGGish feature sliced optimal transport loss model.

since the recording was made in an amateur setting, the target spectrogram contains a noticeable amount of background noise.

The results are surprising. While the generated spectrograms do exhibit vertical structures, they differ significantly from the target. Instead of sharp, well-defined onsets, the generated patterns appear more organic and diffuse, lacking clear black regions that would indicate the absence of energy at certain frequencies. The absence of strong contrasts suggests that the model has not captured the transient nature of the original sound. Despite the shared vertical orientation between the target and the generated output, their overall structure remains highly dissimilar.

C. Perception Changes with VGGish

1) Target: Simple sine wave pattern: Next, we test our modified approach, incorporating additional perception filters alongside the VGGish network. We see the generated Mel spectrograms for the target sine wave spectrogram consists of even smaller white dots than before. Again there is no clear emergence of frequency bands for the sine waves that we would expect. We suspect that our added perception filters are too small in size and are encouraging the model to focus on local texture like features too much. We should make them larger to try capture the broader spectral structure of the target. Sadly this would require an architectural change to our model that is out of the scope of this works time-frame and so we leave this hypothesis to be explored in future work.

2) Target: Clicking sound: Finally, we test the clicking sound again, this time using the VGGish-based model with additional perception filters. The results, shown in Figure 6, produce an aesthetically interesting pattern that in some ways resembles the target. The number of vertical bands is similar, and there is more noticeable separation between them, closer to the onset spacing in the original spectrogram.

However, the spectral detail remains far from accurate. The extra perception filters appear too small to guide the

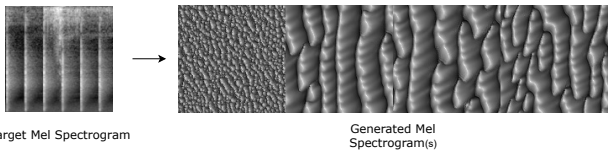

Fig. 6. A target Mel-spectrogram of a stove ignition clicking and the generated spectrograms from the NCA model with our extra perception filters + VGGish feature sliced optimal transport loss model.

model toward a sonically grounded representation, instead encouraging the learning of localized, texture-like patterns. This is evident in the way the vertical "worm-like" bands have their highest intensity at the beginning and gradually fade in a way a bit similar to the target, but lack the sharp, well defined onsets characteristic of the original sound. While the model captures some fine-grained textural features, it struggles with maintaining global coherence, which is crucial for accurate spectrogram representation.

V. Conclusions and Future Work

This work has provided us valuable insights into adapting Neural Cellular Automata (NCA) for generating spectrograms, by building on existing approaches designed for visual textures. Our results highlight the fundamental challenges in transferring techniques from natural images to spectrograms. While both are visually structured, spectrograms encode time-frequency information, meaning that patterns must not only be visually coherent but also sonically meaningful when re-synthesized. This key difference makes direct adaptation difficult, yet our findings offer promising directions for future improvements.

One of our main observations was that using the VGG model in our loss function led to spectrograms that were visually similar to the target but lacked the structure necessary for an accurate audio reconstruction. The introduction of VGGish provided a stronger constraint, producing results that aligned more closely with the target in some limited aspects, particularly in the clicking sound experiments with visible separation between onsets. However, further refinements are needed to ensure that the generated spectrograms don't just vaguely resemble the target visually but also capture the correct spectral and temporal features for meaningful audio synthesis similar to the target.

Our experiments with extra perception filters showed that they contributed to learning localized textural patterns but did not sufficiently enforce global coherence, essential for spectrogram representation. In hindsight, perhaps unsurprising given their small size. Our additional filters appeared to enhance fine-grained details rather than helping to encourage generation similar to the overall spectral structure of the target.

In future work, there are two main areas we would like to improve. Firstly, we need a better suited perception module, such as with long, thin filters that can better capture structures in both the time and frequency domains. Similar filters to this have been explored successfully in audio classification tasks

[12]. This we think would help encourage the NCA model to produce spectrograms that are more sonically accurate to the target since cells will have a wider and sonically useful field of perception. Secondly, we would like to refine our Mel-spectrogram preprocessing pipeline to ensure compatibility between VGGish and BigVGAN. This could involve exploring smaller versions of BigVGAN, training a VGGish-like architecture with the same preprocessing methods as BigVGAN, or experimenting with alternative Mel-spectrogram re-synthesis techniques.

REFERENCES

[1] D. Burraston and E. E. and, "Cellular automata in generative electronic music and sonic art: a historical and technical review," *Digital Creativity*, vol. 16, no. 3, pp. 165–185, 2005, publisher: CAA Website _eprint: https://doi.org/10.1080/14626260500370882. [Online]. Available: https://doi.org/10.1080/14626260500370882

[2] E. R. Miranda, "Granular Synthesis of Sounds by Means of a Cellular Automaton," *Leonardo*, vol. 28, no. 4, pp. 297–300, 1995, publisher: The MIT Press. [Online]. Available: https://muse.jhu.edu/pub/6/article/607044

[3] A. Mordvintsev, E. Randazzo, E. Niklasson, and M. Levin, "Growing Neural Cellular Automata," *Distill*, vol. 5, no. 2, p. e23, Feb. 2020. [Online]. Available: https://distill.pub/2020/growing-ca

[4] E. Niklasson, A. Mordvintsev, E. Randazzo, and M. Levin, "Self-Organising Textures," *Distill*, vol. 6, no. 2, p. e00027.003, Feb. 2021. [Online]. Available: https://distill.pub/selforg/2021/textures

[5] S.-g. Lee, W. Ping, B. Ginsburg, B. Catanzaro, and S. Yoon, "BigVGAN: A Universal Neural Vocoder with Large-Scale Training," Feb. 2023, arXiv:2206.04658 [cs]. [Online]. Available: http://arxiv.org/abs/2206.04658

[6] D. Griffin and J. Lim, "Signal estimation from modified short-time Fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, Apr. 1984, conference Name: IEEE Transactions on Acoustics, Speech, and Signal Processing. [Online]. Available: https://ieeexplore.ieee.org/document/1164317

[7] A. Dorin, "LIQUIPRISM : GENERATING POLYRHYTHMS WITH CELLULAR AUTOMATA," 2002. [Online]. Available: https://api.semanticscholar.org/CorpusID:17964313

[8] J. Serquera and E. R. Miranda, "Evolutionary Sound Synthesis: Rendering Spectrograms from Cellular Automata Histograms," in *Applications of Evolutionary Computation*, C. Di Chio, A. Brabazon, G. A. Di Caro, M. Ebner, M. Farooq, A. Fink, J. Grahl, G. Greenfield, P. Machado, M. O'Neill, E. Tarantino, and N. Urquhart, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 381–390.

[9] J. Armitage, V. Shepardson, and T. Magnusson, "Tölvera: Composing With Basal Agencies," in *Proceedings of the International Conference on New Interfaces for Musical Expression*. Zenodo, Oct. 2024, pp. 282–291. [Online]. Available: https://doi.org/10.5281/zenodo.13904854

[10] L. A. Gatys, A. S. Ecker, and M. Bethge, "Texture Synthesis Using Convolutional Neural Networks," Nov. 2015, arXiv:1505.07376 [cs]. [Online]. Available: http://arxiv.org/abs/1505.07376

[11] 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)*, Mar. 2017, pp. 776–780, iSSN: 2379-190X. [Online]. Available: https://ieeexplore.ieee.org/document/7952261

[12] J. Pons, T. Lidy, and X. Serra, "Experimenting with musically motivated convolutional neural networks," in *2016 14th International Workshop on Content-Based Multimedia Indexing (CBMI)*, 2016, pp. 1–6.