# Audio-reactive Latent Interpolations with StyleGAN

**Hans Brouwer**
Delft University of Technology
hans@wavefunk.xyz

## Abstract

StyleGAN allows for fine-grained control of image generation through its hierarchy of latent and noise inserts. Using musical information like onset envelopes and chromagrams, latent vectors and noise maps can be generated to create interpolation videos that react to music. This paper introduces techniques that create convincing audio-reactive videos by combining multiple interpolations that react to individual musical features. The amount of musical information that can be visually recognized is maximized by mapping different musical features to different parts of the latent and noise hierarchy.

## 1 Introduction

There have been many music videos that use machine learning techniques [3, 11, 12, 17, 22, 23]. However, most either stylize pre-edited clips, suffer from size limitations, or do not visually react to the music. Lee et al. [14] propose a framework that learns to synthesize videos of people dancing from music. However, the architecture relies on domain-specific features and so is limited to dance videos only. Previous audio-reactive videos that use StyleGAN [7, 13, 15, 19, 20] are either focused on syncing the lips with the text, are not able to capture the wide range of elements in the music, or do not make full use of the expressiveness of StyleGAN's architecture.

In this paper, multiple audio-reactive techniques are introduced that better capture the most important aspects of music. We also show how these techniques can be combined while preserving their individual clarity. This increases the amount of musical information that can be encoded visually by leveraging the full expressive capacity of StyleGAN.

The StyleGAN(2) [8, 9] generator uses a hierarchy of convolutional blocks each responsible for different scales of visual information (this is discussed further in Appendix A). Each block is controlled by a latent vector which encodes the semantic structure of that scale and a noise matrix which encodes the stochastic layout of the scale. This multi-scale control of image generation offered by StyleGAN is important for generating convincing audio-reactive interpolations.

Creating a video that reacts well to the music is a process that is unique to each audio source. The artist's task is to find which features in the music can best be mapped to StyleGAN's inputs. Spreading the modulation over different scales prevents the interpolation from becoming too chaotic. Good audio-reactive videos are characterized by how well separate visual movements can be recognized as corresponding to elements of the music. Therefore, having changes in the video which are clearly distinct from each other allows more "bandwidth" to represent the music.

## 2 Extracting Musical Feature Envelopes

Audio-reactive interpolations should correspond to the music both on a short-term scale and long-term scale. Harmonic and rhythmic features can be used to generate recognizable visual changes on short timescales. Mixing together multiple interpolations designed to match each section of the song ensures the global visual narrative fits the musical one, as well.

4th Workshop on Machine Learning for Creativity and Design at NeurIPS 2020, Vancouver, Canada.

**Short-Term Features** The two most important musical features for local reactivity are the chromagram and onset envelope. A chromagram divides the frequency spectrum into categories by pitch (e.g. the 12 notes of the western musical scale). This gives separate envelopes which are high when the given frequencies are being played and low when they are not. The onset envelope peaks when there is a sudden change in the audio spectrum (e.g. when a drum is hit). This captures the rhythmic character of the music.

Often it is desirable to calculate these features for individual instruments, but sometimes it is not possible to find a band of frequencies where the instrument is isolated. One way to alleviate this is by applying harmonic-percussive source separation algorithms [5] and then filtering the separated signals. More recently, neural network based approaches also allow more targeted source separation [4, 6, 21].

**Long-Term Features** On top of these short-timescale features, the longer-term structure of the music can be incorporated. The RMS represents the average energy across the entire spectrum. Louder sections with more instruments will have a higher RMS value while softer sections will be low. Alternatively, algorithms like Laplacian segmentation [16] can be used to analyze the hierarchical pattern structure of music to automatically detect different sections.

## 3 Generating Audio-reactive Interpolations

An expressive audio-reactive interpolation is created by combining multiple sequences of latent vectors or noise maps that react to individual aspects of the music. Video examples showing results can be found at `https://jcbrouwer.github.io/audio-reactive-stylegan-supplement`.

**Latents** Two types of interpolations can be used as building blocks: looping patterns synchronized to a set amount of bars of music and chromagram-weighted sums of a set of latent vectors (discussed further in Appendices B and C, respectively). Using looping patterns tends to work well due to the repetitive nature of music. The chromagram-weighted sum creates a sequence where each note has a unique visual style which blend together based on the notes being played. Using either loops or a chromagram sequence as a base, other interpolations can be added by multiplying the new interpolation by the onset of an instrument and the base interpolation by its inverse. This causes a visual shift towards the instrument's latents whenever it plays. Often a single, static latent is sufficient to mix in by onset, especially for transient envelopes like drums.

**Noise** For noise maps, the exact values have less effect on the resulting image. Rather the standard deviation, spatial location, and smoothness over time are visually apparent. Multiplying a noise sequence with drum onsets will increase its standard deviation whenever the drums hit causing a short, chaotic spasm.

**Mixing** The key to ensuring that different reactive components are recognizable is to map them to different parts of the noise and latent hierarchies. Both hierarchies can be split into about 3 levels: large structure, medium structure/color, and fine detail. On top of this, noise maps have a spatial component and so different parts of the image can be used by different reactive components (e.g. center vs. edges). Sequences designed for each section of a song (e.g. different noise intensities, different latents, faster loops, etc.) can be blended based on long-term features, like RMS, to ensure the video follows the narrative of the music.

**Network Bending** Another way of influencing image generation is by applying transformations to intermediate outputs before passing them to the next layer. This is called network bending [2]. Simple spatial transformations like zooming or rotating applied to lower layers propagate to the output image with a fluid, morphing quality due to the upper layers filling details into the transformed features. Classification networks can be trained to recognize which internal outputs are related to semantic features and transform those individually (e.g. scaling the eyebrows or translating the ears). A related technique called model rewriting [1] works similarly but applies changes to the weights of the generator instead. These techniques both open a wide range of specific targets for audio feature modulation which are clearly distinguishable from latent or noise effects–increasing the musical information that can be conveyed in a given period of time.

## Broader Impact

The techniques described in this paper open new creative pathways towards expressive audio-reactive content. Hopefully this will enable artists in the future to design better accompaniments to their music and convey the messages behind their work more easily.

Generative networks like StyleGAN provide expressive control over high-level structural and textural aspects of images. Where previously artists would have to program a rich generative/procedural system manually, now GANs can be trained to learn representations of arbitrary image data. While the computational requirements to train such GANs are still high, further innovations will continue to lower the barrier to entry. This will allow people without extensive knowledge of visual design programs or music information retrieval to create audio-reactive videos easily.

This might automate away the jobs of today's audio-reactive designers, but it would do so by opening the door for others to create such content more easily. The techniques described in this paper still need to be selected and fine-tuned for different songs or genres for the best results. This means that, for the time being at least, a human in the loop can still provide value over automatic systems.

One interesting consideration that arises from the use of generative networks is the question of how derivative the audio-reactive videos are. A network could, for example, be trained on copyrighted works, causing its outputs to strongly resemble the originals. On the one hand, this could be used for creative remixing of a music video to accompany a remix of the original song, on the other, it might enable profiting off of the visual style of an artist who does not provide consent or receive compensation.

As always there are secondary effects that have the potential to have a negative impact. However, hopefully people will instead focus their effort on using these techniques to create inspiring new works of audio-reactive art.

## Acknowledgments and Disclosure of Funding

Thank you to Lydia Chen, Wiep van der Toorn, Thore Roepman, and Xander Steenbrugge for providing valuable feedback and support.

This work was partially supported by monetary contributions from Subverse bass music collective.

## References

[1] David Bau et al. *Rewriting a Deep Generative Model*. July 30, 2020. arXiv: 2007.15646 [cs]. URL: http://arxiv.org/abs/2007.15646.

[2] Terence Broad, Frederic Fol Leymarie, and Mick Grierson. *Network Bending: Manipulating The Inner Representations of Deep Generative Models*. May 25, 2020. arXiv: 2005.12420 [cs]. URL: http://arxiv.org/abs/2005.12420.

[3] Mike Burakoff and Hallie Cooper-Novack, director. *MGMT - When You Die (Official Video)*. In collab. with Lucia Pier and Jamie Dutcher. Dec. 12, 2017. URL: https://www.youtube.com/watch?v=tmozGmGoJuw.

[4] Alexandre Défossez et al. *Music Source Separation in the Waveform Domain*. arXiv: 1911.13254. URL: http://arxiv.org/abs/1911.13254.

[5] Jonathan Driedger. "Extending Harmonic-Percussive Sepearation of Audio Signals". In: *ISMIR*. 2014.

[6] Romain Hennequin et al. "Spleeter: A Fast and State-of-the-Art Music Source Separation Tool with Pre-Trained Models". In: (2019), p. 2.

[7] Fly Jonathan. *Scatman John - Scatman (Wav2Lip Interpolation Video)*. URL: https://twitter.com/jonathanfly/status/1300976651380109313.

[8] Tero Karras, Samuli Laine, and Timo Aila. *A Style-Based Generator Architecture for Generative Adversarial Networks*. Mar. 29, 2019. arXiv: 1812.04948 [cs, stat]. URL: http://arxiv.org/abs/1812.04948.

[9] Tero Karras et al. *Analyzing and Improving the Image Quality of StyleGAN*. Mar. 23, 2020. arXiv: 1912.04958 [cs, eess, stat]. URL: http://arxiv.org/abs/1912.04958.

[10] Tero Karras et al. *Progressive Growing of GANs for Improved Quality, Stability, and Variation*. Feb. 26, 2018. arXiv: 1710.10196 [cs, stat]. URL: https://github.com/tkarras/progressive_growing_of_gans/blob/35d6c23c578bdf2be185d026c6b3d366c1518120/util_scripts.py#L60.

[11] Mario Klingemann, director. *Automatically Generated BigGAN Music Video*. Dec. 12, 2018. URL: https://www.youtube.com/watch?v=U7Jz17ZZyOg.

[12] Mario Klingemann, director. *Experimental Music Video Made with Neural Face Generator*. Feb. 3, 2017. URL: https://www.youtube.com/watch?v=-elV0lT5-H4.

[13] Mario Klingemann, director. *Hair by Corona™ - StyleGAN2 Generated Music-Reactive Clip*. In collab. with Pitx. Apr. 4, 2020. URL: https://www.youtube.com/watch?v=yWcYcA2HLNc.

[14] Hsin-Ying Lee et al. *Dancing to Music*. Nov. 5, 2019. arXiv: 1911.02001 [cs]. URL: http://arxiv.org/abs/1911.02001.

[15] Robert Luxemburg. *Culture_shock.Py*. URL: https://gist.github.com/rolux/48f1da6cf2bc6ca5833dbacbf852b348.

[16] Brian McFee and Daniel P. W. Ellis. "Analyzing Song Structure with Spectral Clustering". In: *ISMIR*. 2014.

[17] Jana Sam and Alex Mordvintsev, director. *"Neverending Story" Music Video*. In collab. with Cerebral Party and Alice Virt. Jan. 1, 2019. URL: https://www.youtube.com/watch?v=IVBmLRPiQjQ.

[18] *Scipy.Interpolate.Splrep — SciPy v1.5.2 Reference Guide*. URL: https://docs.scipy.org/doc/scipy/reference/generated/scipy.interpolate.splrep.html#scipy.interpolate.splrep.

[19] Xander Steenbrugge, director. *When the Painting Comes to Life...: An Experiment in Visualizing Sound Using AI*. In collab. with Kupla x j'san. Oct. 29, 2019. URL: https://vimeo.com/369531132.

[20] Xander Steenbrugge. *WZRD*. URL: https://wzrd.ai/.

[21] Fabian-Robert Stöter et al. "Open-Unmix - A Reference Implementation for Music Source Separation". In: *JOSS* 4.41 (Sept. 8, 2019), p. 1667. ISSN: 2475-9066. DOI: 10.21105/joss.01667. URL: https://joss.theoj.org/papers/10.21105/joss.01667.

[22] Keijiro Takahashi. *NGX: Pix2Pix VJ Generator & Mixer*. Oct. 4, 2020. URL: https://github.com/keijiro/Ngx.

[23] Pinar & Viola and Sofia Crespo, director. *PANSPERMIA // Music Video for Aski - Se*. In collab. with Dark Fractures and Gene Kogan. Sept. 4, 2019. URL: https://www.youtube.com/watch?v=6Ag_OQ7Ev5Q.

[24] Tom White and Soumith Chintala. *Issue #14 · Soumith/Dcgan.Torch*. URL: https://github.com/soumith/dcgan.torch/issues/14.

## A StyleGAN Architecture

The StyleGAN(2) [8, 9] generator uses blocks of progressively larger style-modulated convolutions with a noise insert at each scale. With an output resolution of 1024x1024 there are 9 distinct scales at which latent vectors and noise matrices are inserted to the network. The lower blocks, which have smaller spatial size, model large-scale structural information while the higher, larger blocks model the details. The latent vector controls high-level structural or textural information (e.g. face shape or hair color in a network trained on FFHQ) while the noise controls the stochastic variation in this information (e.g. the exact pattern of hair or skin pores).

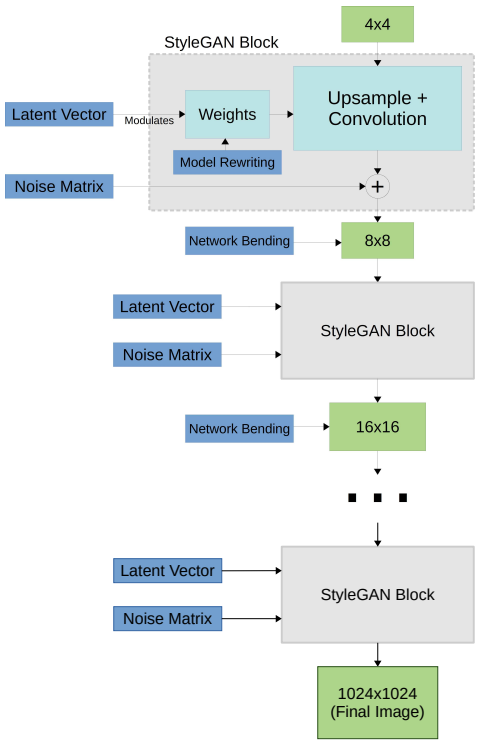


Figure 1: Schematic representation of the structure of the StyleGAN generator. Dark blue boxes represent targets that audio features can control.

Figure 1 shows the structure of the StyleGAN generator. While there are differences between the architectures of StyleGAN1 and StyleGAN2, they do not affect the control points that can be targeted and so the details are left out of the schematic. The green boxes show the values that propagate through the network step by step. The dark blue boxes show the targets that musical features can be mapped to.

While there are 9 sets of 4 of these features in a network of resolution 1024x1024, generally the effects of consecutive blocks will not be distinguishable from each other when being modulated at the same time–especially in the highest blocks. This can vary per trained network, though, so groups of blocks that are visually separable should be evaluated on a case-by-case basis.

## B Looping Latent Sequences

There are multiple ways to generate latent sequences that loop in a visually pleasing way: using spline interpolation, Gaussian filters, or geodesics on the unit n-sphere.

The spline approach is the most straightforward and works well when done in StyleGAN's mapped latent space, W. For a collection of latents, calculate the 1-dimensional spline through each of their coordinates returning to the first latent at the end[18].

The second trick comes from the official implementation of Progressive Growing of GANs [10]. Here a 1-dimensional Gaussian filter is applied along the time dimension of a set of latents. This weighted moving average creates a smooth interpolation. This is useful to make any latent or modulation signal smoother. Note that with higher standard deviations of the Gaussian filter (which give smoother results), the latents are drawn closer to the average latent. Normalizing by the standard deviation of the latents after filtering retains output variety, however, this can cause the final path through latent space to miss the exact latent vectors that were originally filtered.

The unit sphere geodesics approach is less applicable in StyleGAN's W-space as this is a semantically decoupled latent space. However, in the z-space, interpolating along these geodesics tends to give higher quality images than linear or spline interpolation. A discussion of this can be found in a GitHub issue in the DCGAN repository [24].

However the loops are generated, setting their length so that they repeat in time to patterns in the music helps create the feeling that the video matches the audio. While this trick relies on the human tendency to seek connections even when there are none, mixing a few other audio-reactive sequences into a base sequence that is simply looping creates convincing audiovisual interpolations.

## C Chromagram-weighted Latent Sequences

A chromagram-weighted latent sequence can be created by taking the dot product of a normalized chromagram and a tensor of latent vectors. When the chromagram is normalized to sum to one at each time step, the result is an average of the latent vectors weighted by the spectral energy in the frequencies of each note.

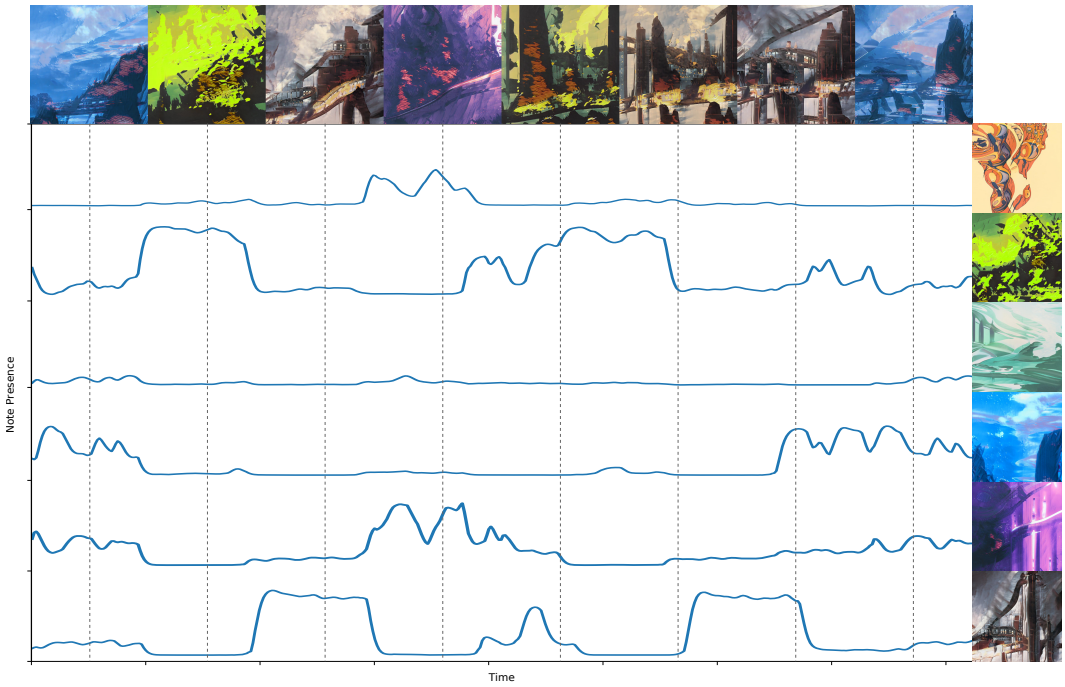


Figure 2: A 6-bin chromagram's magnitudes are plotted in the center. The images corresponding to the latent vectors of each bin are along the right side. The chromagram-weighted sequence is along the top–the sum of each chromagram bin times its latent vector. The dotted lines show the point in time that each frame occurs.

Figure 2 shows the magnitude of each bin of an example 6-bin chromagram over time. On the right are the images which each latent vector produces and along the top are is the chromagram-weighted sequence. When the magnitude of the bin is high, its corresponding latent vector is blended into the image more strongly. Due to StyleGAN's learned, semantically decoupled latent space, averaging latent vectors produces images that have traits of each of the latent vectors that are mixed in.

Humans can easily hear when different chords or notes are played and so using latent sequences that change according to these tones is a way to ensure that the video's movements can clearly be recognized as stemming from the music.

## D Post-processing Envelopes

Often signals such as onset or chromagrams calculated from audio will be noisy and unstable. This can cause unpleasant visual jitter or lead to visual changes which are less/more emphatic than their audio counterparts. While using better source separation or filtering the audio can help, there are a couple of ways to post-process these envelopes that can make a big difference as well.

The best audio-reactive results come from carefully tweaking combinations of these post-processing effects on each musical feature's envelope.

**Filtering** A moving average or Gaussian filter (as discussed in Appendix B) helps eliminate jitter from envelopes. This reduces the dynamic range of the envelope so should usually be followed with a normalization step to compensate. For transient envelopes, the second half of the filter kernel can be reduced (or set to zero) to make the filter more causal (although the kernel must be normalized to sum to one again). This smooths the signal without looking forward in time. This helps prevent visual transients from starting to ramp up before the corresponding audio transient has occurred.

**Compression and Expansion** Another important set of effects are expansion, compression, and clipping. These effects work differently on the signal based on its value. For an example of expansion, a threshold can be set at 0.5, above which the signal is multiplied by 2 and below which the signal is divided by 2 (again, usually this is followed by normalizing the signal back to between 0 and 1). This creates a more extreme envelope which is only high when the musical feature is strongly present.

Flipping the multiplication and division to the opposite sides of the threshold has the opposite effect: compressing the envelope's dynamic range. This is useful when the visual reaction is not strong enough compared to the audio.

Another version of compression is to simply multiply the entire signal and then clip any values above a threshold to that maximum. This creates flattened peaks which can be especially useful for long term envelopes (e.g. for mixing in latent sequences for different sections).